



**Universidade de
Aveiro**

Departamento de Electrónica,
Telecomunicações e Informática

2010

**Vera Marisa Martins
Teixeira**

DSD-mobile

Interface móvel para portal departamental



**Vera Marisa Martins
Teixeira**

DSD-mobile

Interface móvel para portal departamental

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e de Telecomunicações (Mestrado Integrado), realizada sob a orientação científica do Professor Doutor António Luís Jesus Teixeira, Professor Associado e da Professora Doutora Maria Beatriz Alves de Sousa Santos, Professora Associada com Agregação do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro.

o júri

presidente

Prof. Doutor Joaquim Arnaldo Carvalho Martins

Professor Catedrático do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

orientador

Prof. Doutor António Luís Jesus Teixeira

Professor Associado do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro (orientador)

co-orientadora

Prof^a. Doutora Maria Beatriz Alves de Sousa Santos

Professora Associada com Agregação do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro (co-orientadora)

arguente

Prof. Doutor António Miguel Pontes Pimenta Monteiro

Departamento de Engenharia Informática da Faculdade de Engenharia da Universidade do Porto

agradecimentos

Em primeiro lugar, quero deixar o meu agradecimento aos meus orientadores, Professor António Luís Jesus Teixeira e Professora Maria Beatriz Alves de Sousa Santos, pela compreensão e disponibilidade ao longo deste trabalho.

Ao colega Francisco Taborda pela constante paciência e ajuda nos assuntos técnicos.

Aos meus amigos pelo apoio na realização deste trabalho e pela constante companhia.

A todos os colegas que contribuíram activamente na avaliação da aplicação desenvolvida, um obrigada pela disponibilidade.

Aos meus pais, que sempre me apoiaram e incentivaram a ultrapassar os obstáculos da vida.

Um obrigada aos meus irmãos pelo apoio constante ao longo destes anos e disponibilidade para tudo. Em especial ao Leandro, por existir.

Ao meu namorado Ricardo, pela incondicional ajuda e companhia em todos os momentos. Pela disponibilidade e incentivo, um especial agradecimento.

A todos o meu muito Obrigada.

palavras-chave

Dispositivos móveis, Sistemas *Web*, Usabilidade, Aplicações *Web Mobile*

resumo

Os dispositivos móveis permitem aceder à informação em qualquer lugar e em qualquer momento pois, fica imediatamente disponível a quem necessita dela.

Os estudantes e docentes de uma Universidade são um grupo de utilizadores que fica potencialmente beneficiado com a evolução das tecnologias móveis. O acesso facilitado a toda a informação torna-se real através do acesso por dispositivos móveis.

Criar uma aplicação *Web* constitui um desafio, pois não basta conhecer os processos realizados pelos potenciais utilizadores, sendo igualmente necessário assegurar que funcionam do mesmo modo quando são executados através de um equipamento com características muito próprias, como é o caso dos telemóveis.

O trabalho realizado no contexto desta dissertação, teve como principal finalidade a criação de uma interface móvel do portal departamental dsd.av.it.pt capaz de fornecer aos alunos do Departamento de Electrónica, Telecomunicações e Informática, a possibilidade de acesso a informações necessárias no momento. A dissertação em causa passou, numa primeira fase, por uma investigação do estado da arte no contexto dos dispositivos móveis. Foi realizada a análise de requisitos dos utilizadores de forma a perceber as suas necessidades pois a capacidade e mentalidade dos utilizadores são por vezes um obstáculo à implementação das tecnologias móveis. A criação de uma aplicação enfrenta não só o desafio na concepção e na implementação mas também na aceitação por parte dos seus utilizadores. Seguidamente foi implementada a dita aplicação tendo em consideração os constrangimentos existentes nos dispositivos móveis que pudessem afectar a consistência e coerência da mesma. Por fim, foram realizados testes, através de um protótipo funcional, com utilizadores reais de forma a avaliar a validade da solução proposta.

keywords

Mobile devices, Web Systems, Usability, Web Applications Mobile Web

abstract

The mobile devices enable the access to information at any place and at any time since this is immediately available to those who need it.

The students and lecturers of a university are a group of users which is potentially benefited with the development of mobile technologies. The easy access to all information becomes real through the access through mobile devices.

Creating a Web application is a challenge since it is not enough to know the processes carried out by potential users, being also necessary to guarantee they work the same way when they are executed through equipment with own characteristics, as the case of mobile phones.

The work done in the scope of this dissertation had as main aim the creation of a mobile interface of the departmental portal dsd.av.it.pt able to provide the students of the Department of Electronics, Telecommunications and Informatics, with the possibility of access to needed information. This dissertation went in a first phase through an investigation of the state of art in the scope of mobile devices. The analysis of the requirements of users was carried out in order to understand their necessities since the ability and mentality of the users are many times an obstacle to the implementation of mobile technologies. The creation of an application faces not only the challenge in the conception and implementation but also on the acceptance by its users. Then this application was implemented taking into account the existing constraints in the mobile devices that could affect its consistency and coherence. At last, tests were carried out through a functional prototype, with real users in order to assess the validity of the proposed solution.

Índice

Índice	xiii
Índice de Figuras	xv
Índice de Tabelas	xvii
Acrónimos	xix
1 Introdução	1
1.1 Motivação	1
1.2 Objectivos	2
1.3 Enquadramento	2
1.4 Estrutura da dissertação	3
1.5 Principais contribuições	4
2 Estado da Arte	5
2.1 Desenvolvimento Mobile Web	5
2.1.1 Evolução dos dispositivos móveis	5
2.1.2 Convergência da <i>Web</i> com os dispositivos móveis	8
2.1.3 <i>Mobile 2.0</i>	25
2.1.4 <i>Mobile Web design</i>	30
2.2 Tecnologias	43
2.2.1 <i>Client-Side</i>	45
2.2.2 <i>Hosting</i>	48
2.2.3 SGBD	49
2.2.4 Páginas <i>Web</i> dinâmicas	52
2.2.5 Linguagens dinâmicas	54
2.2.6 MVC	55
2.2.7 <i>Frameworks</i>	58
2.3 Avaliação da Usabilidade de Sistemas Móveis	60
2.3.1 Requisitos	61
2.3.2 Análise de requisitos	62
2.3.3 Usabilidade	68
2.3.4 Soluções centradas no utilizador	71
2.3.5 Aplicação aos dispositivos móveis	74
3 Plataformas DSD	79
3.1 DSD Web – A plataforma existente	79
3.1.1 Objectivos da plataforma	79
3.1.2 Funcionalidades <i>Front Office</i> (Aluno)	79
3.1.3 Tecnologias	83
3.1.4 Conclusões	84
3.2 DSD-mobile	84
3.2.1 Contexto do projecto	84
3.2.2 Aquisição e análise de requisitos	85
3.2.3 Aplicação Web versus Nativa	92
3.2.4 Design	93
3.2.5 Tecnologias de implementação	100
3.2.6 Testes à aplicação	101

4	Avaliação de Usabilidade da aplicação	107
4.1	Métodos utilizados.....	107
4.1.1	Avaliação Heurística.....	108
4.1.2	Teste de usabilidade.....	110
4.1.3	Conclusões.....	116
5	Conclusões gerais	119
5.1	Trabalho Futuro	120
6	Referências bibliográficas	123
7	Anexos.....	129
7.1	Anexo A.....	129
7.1.1	Questionário preliminar para análise de requisitos.....	129
7.2	Anexo B.....	135
7.2.1	Lista de tarefas de utilizadores	135
7.2.2	Guião do observador.....	137
7.2.3	Questionário final	139

Índice de Figuras

Figura 1 - Evolução dos dispositivos móveis (adaptado de (Steinbock 2005))	6
Figura 2 - Modo de interacção <i>focus</i> (retirado de Google).....	15
Figura 3 - Modo de interacção por cursor (retirado de Google).....	16
Figura 4 - Modo de interacção <i>multitouch</i> (retirado de Google).....	16
Figura 5 - Modo de interacção <i>touch</i> (retirado de Google)	16
Figura 6 - Página Web "ThePhoneHouse", na sua versão <i>desktop</i> (à esquerda) e versão <i>mobile</i> (à direita).....	24
Figura 7 - Página Web "KLM", na sua versão <i>desktop</i> (à esquerda) e versão <i>mobile</i> (à direita).....	24
Figura 8 - Sete princípios que colocam a experiência colectiva no centro da Web Inteligente (adaptado de (Firtman 2010)).....	27
Figura 9 - Comparação entre os vários tamanhos de ecrã (adaptado de (Mehta 2008))	31
Figura 10 - Fluxo típico da informação em páginas Web de <i>desktop</i> (adaptado de (Mehta 2008))	32
Figura 11 - Fluxo de informação desejado em páginas exibidas em dispositivos móveis (adaptado de (Mehta 2008))	33
Figura 12 - Página Web "The Onion" versão <i>desktop</i> (à esquerda) e versão <i>mobile</i> (à direita).....	34
Figura 13 - Exemplo dos diferentes níveis de <i>posterization</i> que podem ocorrer em dispositivos com diferentes profundidades de cor (retirado de (Fling 2009)).....	36
Figura 14 - Exemplos da letra "a" serifada e não-serifada (retirado de Google).....	38
Figura 15 - <i>ClearType</i> inactivo (adaptado de (Microsoft 2010c)).....	39
Figura 15 - <i>ClearType</i> activo (adaptado de (Microsoft 2010c))	39
Figura 17 - Exemplo de um Sistema de Gestão de Bases de Dados (adaptado de (Cardoso 2009))	50
Figura 18 - Esquema ilustrativo de um pedido ao servidor de uma página Web (adaptado de (Darie e Barnett 2008)).....	53
Figura 19 - Diagrama explicativo do padrão MVC (adaptado de (Microsoft 2010a)).....	57
Figura 20 - Processo geral para a análise de requisitos (adaptado de (Maguire e Bevan 2002)))	62
Figura 21 - Processo para o desenvolvimento de sistemas interactivos centrados no utilizador (adaptado de (Teixeira 2008))	73
Figura 22 - Arquitectura da plataforma DSD (adaptado de (Mota 2009))	83
Figura 23 - Resultado do inquérito correspondente ao tipo de telemóvel que o inquirido possui.....	87
Figura 24 - Resultado do inquérito correspondente à utilização de Internet no telemóvel .	87
Figura 25 - Resultado do inquérito quanto à frequência de utilização da plataforma DSD Web	88
Figura 26 - Resultado do inquérito quanto à utilidade da aplicação DSD-mobile	90
Figura 27 - Sugestões fornecidas pelos inquiridos para a aplicação <i>mobile</i>	91
Figura 28 - Comparação entre o fluxo de informação desejável e o fluxo existente na aplicação DSD-mobile.....	94

Figura 29 - Ausência de ícones no menu da plataforma <i>Web</i>	96
Figura 30 - Ícones utilizados no menu da aplicação DSD-mobile	97
Figura 31 - Diferentes tipos de botões existentes na aplicação DSD-mobile.....	98
Figura 32 - Exemplo do esquema de navegação <i>breadcrumb</i> na aplicação.....	99
Figura 33 - Nível de facilidade por tarefa.....	113
Figura 34 - Antes e depois de efectuar <i>login</i>	115
Figura 35 - Diferença entre como sair de uma área do menu principal e sair da área de perfil	116

Índice de Tabelas

Tabela 1 - Previsão de venda de dispositivos móveis com os diferentes Sistemas Operativos (adaptado de (Gartner 2010))	10
Tabela 2 - Características dos <i>browsers</i> dos dispositivos móveis (adaptado de (Firtman 2010))	17
Tabela 3 - Cores suportadas por diferentes dispositivos móveis (adaptado de (Fling 2009))	37
Tabela 5 - Simplificação do código JavaScript utilizando sintaxe JQuery	59
Tabela 6 - Factores de contexto de utilização (adaptado de (Maguire e Bevan 2002))	64
Tabela 7 - Funcionalidades <i>Front Office</i> (adaptado de (Mota 2009))	82
Tabela 8 - Dispositivos físicos utilizados para teste da aplicação	102
Tabela 9 - Simuladores utilizados nos testes da aplicação	103
Tabela 10 - Grau de gravidade de cada uma das heurísticas	110
Tabela 11 - Resultados do teste de usabilidade	112
Tabela 12 - Valores acerca de aspectos gerais da aplicação.....	114
Tabela 13 - Valores acerca de aspectos específicos da aplicação	114

Acrónimos

A

AJAX	<i>Asynchronous JavaScript and XML</i>
API	<i>Application Programming Interface</i>
ASP	<i>Active Server Pages</i>
APR	<i>Apache Portable Runtime</i>

C

CSS	<i>Cascading Style Sheets</i>
CPU	<i>Central Processing Unit</i>
CLR	<i>Common Language Runtime</i>

D

DSD	Distribuição de Serviço Docente
DETI	Departamento de Electrónica, Telecomunicações e Informática
DHTML	<i>Dynamic Hipertext Markup Language</i>
DSDM	<i>Dynamic Systems Development Method</i>
DOM	<i>Document Object Model</i>

G

GPL	<i>General Public Licence</i>
-----	-------------------------------

H

HTML	<i>Hipertext Markup Language</i>
HD	<i>High Defenition</i>
HTTP	<i>HyperText Transfer Protocol</i>

I

ISO	<i>International Organization for Standardization</i>
IE	<i>Internet Explorer</i>
IDE	<i>Integrated Development Environment</i>
IIS	<i>Internet Information Server</i>

	ISAPI	<i>Internet Server Application Programming Interface</i>
J		
	JDE	<i>Java Development Environment</i>
	JSON	<i>JavaScript Object Notation</i>
	JSP	<i>JavaServer Pages</i>
	JVM	<i>Java Virtual Machine</i>
M		
	MMS	<i>Multimedia Messaging Service</i>
	MDS	<i>Mobile Data Service</i>
	MVC	<i>Model-View-Controller</i>
	MIT	<i>Massachusetts Institute of Technology</i>
O		
	OS	<i>Operating System</i>
P		
	PC	<i>Personal Computer</i>
	PDA	<i>Personal Digital Assistant</i>
	PPI	<i>Pixels Per Inch</i>
	PL/SQL	<i>Procedural Language/Structured Query Language</i>
	PHP	<i>Hypertext Preprocessor</i>
R		
	RAD	<i>Rapid Application Development</i>
	RIM	<i>Research In Motion</i>
	RAM	<i>Random Access Memory</i>
S		
	SMS	<i>Short Messaging Service</i>
	SDK	<i>Software Development Kit</i>
	SGML	<i>Standard Generalized Markup Language</i>
	SGBD	<i>Sistema de Gestão de Base de Dados</i>
	SOA	<i>Service Oriented Architecture</i>

U

URL *Uniform Resource Locator*

V

VGA *Video Graphics Array*

W

WWW *World Wide Web*

WAP *Wireless Application Protocol*

WML *Wireless Markup Language*

X

XML *Extensible Markup Language*

XHTML MP *eXtensible Hypertext Markup Language Mobile Profile*

XSLT *eXtensible Stylesheet Language for Transformation*

1 Introdução

1.1 Motivação

Dado o número crescente de alunos nas Universidades, estas tiveram forçosamente que evoluir de forma a responder às necessidades exigidas. Com este crescimento teve de se aumentar o número de docentes e deste modo foi necessário organizar as diferentes tarefas de gestão e manutenção das respectivas instituições de uma forma menos complexa. É neste sentido que surge a plataforma DSD.

Criada com o intuito de servir como instrumento de interacção com toda a comunidade académica, mais concretamente no que diz respeito à gestão de tempo e espaço, a plataforma DSD veio minimizar a complexidade previamente à sua existência (Mota 2009).

A ideia de implementar parte das principais funcionalidades da plataforma DSD para ambiente *mobile* é assente na elevada predisposição que, actualmente a maior parte dos jovens tem de utilizar os dispositivos móveis com vista a realizar determinadas tarefas sem a necessidade da utilização do computador.

Um estudo efectuado pelo Grupo Marktest (conjunto de empresas especializado na área de estudos de mercado e processamento de informação) que data de Fevereiro de 2010, mostra que a taxa de penetração dos telemóveis em Portugal já está acima dos 90%. Este estudo indica também que esta “taxa de penetração do serviço móvel é superior junto dos jovens, atingindo valores acima dos 99% junto dos que têm entre 15 e 34 anos” (Marktest 2010).

Com o crescimento da capacidade de processamento dos dispositivos móveis, actualmente é possível aceder a qualquer tipo de informação na rede através de um telemóvel. Na sua maioria, os dispositivos já têm um *Web browser* embutido que permite realizar esta tarefa mais facilmente.

É conciliando estes factores que surge este trabalho. Serão feitos estudos para saber quais as funcionalidades pertinentes a adaptar para dispositivos móveis.

“Mobile phones are cheaper than PCs, there are three times more of them, growing at twice the speed, and they increasingly have Internet access. What is more, the World Bank estimates that more than two-thirds of the world’s population lives within range of a mobile phone network. Mobile is going to be the next big Internet phenomenon. It holds the key to greater access for everyone – with all the benefits that entails” (Schmidt 2006).

1.2 Objectivos

O principal objectivo deste trabalho centra-se na construção de uma aplicação baseada em *Web browser* capaz de ser acedida através de dispositivos móveis. Pretende-se adaptar as principais funcionalidades da plataforma DSD existente na Universidade de Aveiro de modo a serem acedidas através de um telemóvel com ligação à Internet.

Para que este desenvolvimento aconteça de forma eficiente é necessário:

- Analisar o contexto e as necessidades da distribuição de serviço docente, bem como sistemas já existentes;
- Estudar os requisitos dos vários utilizadores da plataforma existente; estudar as diferenças entre o acesso à Internet através de computadores e de dispositivos móveis no que diz respeito a limitações;
- Estudar as soluções tecnológicas existentes que melhor se adaptem à solução pretendida;
- Implementar a solução através de uma interface usável e, por fim, realizar os testes necessários com recurso a uma amostra representativa do público-alvo.

1.3 Enquadramento

Enquadrada nas principais actividades de gestão do DETI, a plataforma DSD *Web* existente surge para responder às necessidades dos diferentes perfis de utilizadores, alunos, docentes e funcionários não docentes.

Com a crescente exigência de mobilidade imposta pelo mundo profissional, torna-se comum a necessidade de ferramentas que permitam trabalhar em qualquer lugar. Ferramentas como PDAs (*Personal Digital Assistant*), *smartphones* ou ainda *tablets* PC's tornam-se assim imprescindíveis, uma vez que, na sua maioria, permitem ligação a redes *wireless* e 3G.

Por estes factores os sistemas móveis têm vindo a ser a solução encontrada para aceder a um número significativo de serviços tanto para actividades pessoais como profissionais.

O projecto *DSD-Mobile* surge então para facilitar o acesso à plataforma *DSD Web* a quem tem um telemóvel com possibilidade de ligação à Internet.

1.4 Estrutura da dissertação

Esta dissertação está dividida em cinco capítulos principais.

No capítulo 2 é feita uma análise do estado da arte relativamente ao desenvolvimento *Web* para dispositivos móveis, ou seja, tendo como base o *Web browser*. É realizado um estudo para que se verifique a portabilidade do sistema entre diferentes *Web browser*, assim como, um estudo no que diz respeito às tecnologias envolvidas no desenvolvimento de aplicações deste tipo. Por fim, é feita uma análise do estado da arte relativa à avaliação de usabilidade de sistemas móveis.

No capítulo 3 é apresentada uma descrição da plataforma *DSD* existente (parte *Front Office* do actor aluno), no que concerne aos seus objectivos, às funcionalidades e às tecnologias envolvidas no seu desenvolvimento, bem como uma descrição do trabalho efectuado no âmbito desta dissertação. É abordada a metodologia utilizada, os requisitos necessários à construção do sistema, as opções de implementação e por fim as tecnologias utilizadas.

No capítulo 4 é descrita a avaliação do sistema final. É descrita a metodologia adoptada, analisados os resultados e tiradas conclusões quanto ao sistema implementado.

Por último, no capítulo 5 apresentam-se as conclusões gerais relativamente a problemas encontrados e identificado o trabalho que será necessário realizar no futuro.

1.5 Principais contribuições

Na opinião da autora, as principais contribuições desta dissertação prendem-se com:

- O estudo do estado da arte actual na área do desenvolvimento *Web mobile*;
- A análise de requisitos dos utilizadores da plataforma DSD *Web*, de forma a detectar e compreender as suas necessidades para com o sistema DSD-mobile.
- A familiarização com a plataforma *Web* existente, assim como a análise das suas funcionalidades, no que diz respeito à componente *front office* do utilizador com estatuto de aluno.
- A implementação de um protótipo funcional da aplicação *Web* DSD-mobile.
- O teste da aplicação, bem como a avaliação da usabilidade do sistema final com utilizadores.

2 Estado da Arte

Tendo como objectivo desenvolver, de forma adequada, funcionalidades e tarefas para uma qualquer plataforma, é necessário analisar o estado da arte actual. No caso deste estudo, esta análise abrange o estado da arte relativo ao desenvolvimento *Web* para dispositivos móveis. É dada ênfase à temática *mobile Web*, relacionada com o ambiente de execução de aplicações *Web* através de dispositivos móveis, ao *mobile 2.0* e ao aproveitamento da experiência colectiva dos utilizadores.

Esta análise abrange não só o estudo das tecnologias inerentes ao processo de desenvolvimento deste tipo de aplicações *Web*, mas também o estudo da usabilidade das mesmas.

2.1 *Desenvolvimento Mobile Web*

2.1.1 *Evolução dos dispositivos móveis*

Os telemóveis são o maior segmento de dispositivos móveis, apresentando-se nas mais variadas formas e feitios. Geralmente, são construídos com componentes como memória e processador desenhado especificamente para os sistemas operativos que irão ter instalados. Symbian, *Windows Mobile*, Android, *Mac OS X*, BlackBerry, *Palm*, *Linux*, são alguns exemplos de sistemas operativos utilizados em diferentes gamas de dispositivos móveis. Alguns destes sistemas são desenvolvidos para dispositivos específicos como, por exemplo, o caso do Apple iOS que é desenvolvido especificamente para iPhone e iPad.

O objectivo primordial de um telemóvel é falar, mas a sua evolução tornou possível a realização de tarefas mais complexas. Passamos a ter telefones inteligentes em vez de telemóveis com recursos simples e limitados (Fling 2009). Isto resultou da existência de uma vasta variedade de dispositivos no mercado.

Apesar de ainda existirem milhões de dispositivos monocromáticos, os telemóveis com ecrãs a cores são agora uma realidade. O tamanho do ecrã, resolução e

cores variam muito. Existem ecrãs de 120x120 *pixels*, 320x240 *pixels* e até superiores. Normalmente o tipo de mecanismo de entrada é um teclado, sendo que alguns telefones têm ainda teclas adicionais ou um *joystick* de ajuda à navegação. Modelos com ecrãs maiores normalmente possuem um sistema de entrada baseado em *stylus* (uma espécie de caneta com dimensões mais reduzidas) ou teclados tácteis directamente no ecrã (Steinbock 2005).

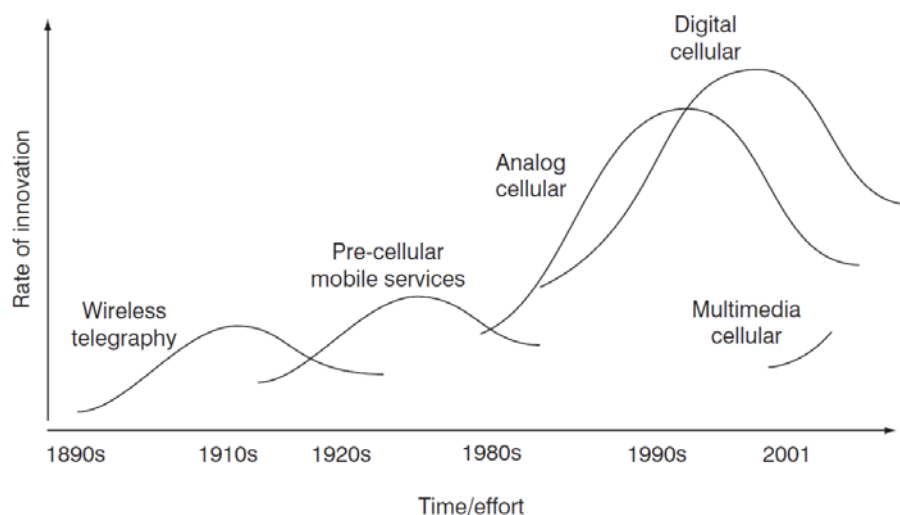


Figura 1 - Evolução dos dispositivos móveis (adaptado de (Steinbock 2005))

As SMS (*Short Message Service*) revolucionaram a utilização do telemóvel servindo agora para todo o tipo de negócios e/ou entretenimento. Posteriormente surgiram as MMS (*Multimedia Message Service*) que permitem enviar ficheiros de música, fotos ou vídeos. No entanto, a câmara foi a funcionalidade que mais alterou a forma como os dispositivos móveis são utilizados. Inicialmente com imagens de qualidade VGA (*Video Graphics Array*) (640 x 480 *pixels*), vão agora até resoluções que chegam aos 5 *Megapixels*, flash e *software* específico que permite editar as fotografias directamente no telemóvel. Prova desta utilização massiva são os milhões de vídeos amadores realizados através de um telemóvel que se encontram na Internet (Marques 2010).

Mas isto não é tudo. Actualmente, os telemóveis são, por si só, pequenos computadores, para os quais se pode desenvolver aplicações que aumentem as suas funcionalidades ou adicionar novos recursos.

Cada telemóvel vem com um conjunto de funcionalidades e aplicações, mas existem milhares de programadores que criam as suas próprias aplicações. O Android da *Google* permite até mesmo substituir as aplicações base (*built-in*), dando ao programador total controlo sobre o dispositivo (Google 2010b).

Actualmente, a maior parte dos telefones são distribuídos com algum tipo de *Web browser* pré-instalado. O *Opera mini* é o mais popular, há já alguns anos (LIEN 2010). Como diferentes *browsers* suportam diferentes funcionalidades, os testes de compatibilidade entre os vários *browsers* é o maior desafio do desenvolvimento *Web* móvel. Segundo Mirav Mehta (2008) a situação é muito pior que nos *browsers desktop*, no que diz respeito ao desenvolvimento *Web*, devido às suas limitações.

Existe ainda outra classe de dispositivos, os PDA's, que foram utilizados durante muito tempo e têm evoluído gradualmente. Nestes dispositivos é disponibilizado, maioritariamente, *software* de produtividade – e-mail, ferramentas *Office* e *software* personalizado. Normalmente têm um teclado QWERTY, sendo a funcionalidade de telefone um extra para os PDAs's. A maior parte dos consumidores de PDA's são agora consumidores de *smartphones* (Exame Informática 2007).

Geralmente os telemóveis eram utilizados para comunicações de voz, mas actualmente a sua utilização para comunicação de dados está a crescer rapidamente.

Segundo relatórios da consultora Chetan Sharma (2008), o Brasil, os Países Baixos e regiões da República Checa têm mais de 30% das suas receitas em dados móveis. “Embora grande parte destes dados sejam SMSs, outros serviços estão a ganhar terreno. Os EUA (Estados Unidos da América) têm entre 50%-60% de receitas em serviços, excluído SMS's, enquanto o Japão e a Coreia rondam valores entre os 70-75”. Esta tendência sugere que há já uma forte utilização de aplicações *Web*.

Da mesma forma, estudos da empresa Informa Telecoms & Media (2007), sediada em Londres, mostram que o número de subscrições de Internet móvel subirá para quatro biliões em 2011. Isto significa que as pessoas estão, cada vez mais, a utilizar os telemóveis para ter acesso a todo o tipo de informação. Esta possibilidade de acesso à informação em qualquer altura e em qualquer lugar, em que haja cobertura de rede é, desde já, uma grande vantagem da *mobile Web*.

2.1.2 Convergência da Web com os dispositivos móveis

De acordo com Brian Fling (2009), actualmente o aproveitamento das potencialidades da *Web* integradas com os serviços disponibilizados pelos dispositivos móveis, é o futuro do desenvolvimento móvel. O aparecimento de dispositivos como, por exemplo, o iPhone ou da vasta gama de *smartphones* no mercado é a grande prova deste facto. Com o aparecimento deste tipo de dispositivos o consumo da *mobile Web* aumentou significativamente.

2.1.2.1 O browser como ambiente de execução de aplicações

Quando acedemos à Internet através de um telemóvel, o contexto de utilização pode ser bastante diversificado. Ou porque estamos a viajar, à procura de um local próximo, à espera de alguma coisa, a resolver alguma tarefa ou simplesmente por diversão. Por isso, no desenvolvimento de aplicações *Web mobile* é necessário ter especial atenção ao facto de que os dispositivos móveis não são utilizados da mesma forma que os computadores.

Enquanto realizamos alguma tarefa num computador, temos um grande ecrã que pode mostrar uma grande quantidade de informação. Estamos habituados a ver muita informação ao mesmo tempo e a “varrer” as páginas *Web* rapidamente e em seguida seleccionar algo que nos é útil. Temos, muitas vezes, acesso a Internet banda larga, podemos executar ficheiros de música e vídeo ou até ter vários separadores com várias páginas de Internet abertas ao mesmo tempo. Ou seja, realizar inúmeras tarefas quase em simultâneo. A esta capacidade dá-se o nome de multiprocessamento, ou seja, a partilha do tempo de utilização do processador entre vários processos (Silberschatz, Galvin et al. 2010).

No contexto da *World Wide Web*, a interacção com o utilizador é feita através de programas de interface genéricos, comumente chamados de *browsers*. Estes, surgiram com o intuito de permitir ler e navegar entre páginas distribuídas pela rede. Em suma, um *browser* é um programa de computador que possibilita a interacção dos utilizadores com documentos virtuais da Internet – páginas *Web* – que estão de alguma forma hospedadas num servidor *Web* (Kayne 2010).

Se o futuro das comunicações móveis é a *Web*, então isto só faz sentido se o ambiente de execução de aplicações for o *browser*.

Os *Web browsers* concebidos para dispositivos móveis, comportam actualmente, muitas das capacidades dos *browsers desktop*. No entanto, são diferentes em certos aspectos, como por exemplo: para apresentar páginas de Internet em pequenos ecrãs, a página tem que ser ajustada ao tamanho do ecrã do dispositivo; o *browser* deve ser eficiente ao ponto de não sobrecarregar demasiado a memória devido ao limitado poder de processamento; deve também, ser facilmente controlado por um dispositivo de entrada simples, como por exemplo, o teclado para introdução de texto (Cotton e Commarford 2005).

Existe uma variedade de *browsers* disponíveis especificamente para os dispositivos móveis. Alguns são optimizados para dispositivos com recursos muito limitados, como ecrã pequeno e pouco poder de processamento. No entanto, o melhor dos *browsers* pode mostrar a maioria das páginas *Web* oferecendo *zoom* e teclas de atalho enquanto outros apenas exibem *sites* optimizados para os dispositivos móveis.

O que se quer, na verdade, é garantir que o acesso à Internet através de telemóveis ofereça uma experiência similar para o utilizador como quando acede através de um computador.

Embora muitos dispositivos não ofereçam muita escolha relativamente ao *browser* a utilizar, actualmente, a maior parte dos dispositivos permite escolher um conjunto de *browsers* que são compatíveis com o sistema operativo em questão. Mesmo assim a situação é mais complexa que no mundo *desktop*. Praticamente todos os dispositivos vêm com um *browser* pré-instalado e muito poucos podem ser desinstalados ou actualizados. Existem excepções, como o caso dos sistemas iOS, WebOS e Android em que os *browsers* são automaticamente actualizados quando o utilizador faz a actualização ao *firmware* do sistema (Firtman 2010).

Os *browsers* pré-instalados

Existe menos variedade de *browsers* que de dispositivos, o que torna a sua análise facilitada. Nesta secção são abordadas algumas características dos *browsers* pré-instalados nos dispositivos com os sistemas operativos mais utilizados.

Sistema Operativo	2009	2010	2011	2014
Symbian	46.9	40.1	34.2	30.2
Android	3.9	17.7	22.2	29.6
BlackBerry	19.9	17.5	15.0	11.7
iOS	14.4	15.4	17.1	14.9
Windows Phone	8.7	4.7	5.2	3.9
Outros Sistemas	6.1	4.7	6.3	9.6

Tabela 1 - Previsão de venda de dispositivos móveis com os diferentes Sistemas Operativos (adaptado de (Gartner 2010))

De acordo com uma notícia da Gartner, Inc. (Gartner 2010), o Android tornar-se-á o segundo sistema operativo mais utilizado em 2010 desafiando o primeiro lugar do sistema Symbian em 2014. O que indica que o mercado dos sistemas operativos móveis será dominado pelos sistemas Symbian e Android. Estes dois sistemas estarão presentes nos dispositivos móveis numa taxa de 59,8% até 2014. O Symbian permanecerá no topo do *ranking* devido ao preço competitivo dos dispositivos em que se encontra instalado (na maioria Nokia). No entanto, até ao final do período de previsão (2014), o primeiro lugar será disputado entre Symbian e Android que estarão num nível muito semelhante.

“O mercado mundial de Sistemas Operativos móveis é dominado por quatro referências: Symbian, Android, RIM (Research In Motion) e iOS. O lançamento de Sistemas Operativos actualizados como iOS 4, BlackBerry OS 6, Symbian 3 e 4 e Windows Phone 7, irão ajudar a manter um forte crescimento na venda de smartphones e estimular a inovação”

(Roberta Cozza, analista principal de pesquisas da Gartner, Inc.)

Então, de acordo com estas declarações, faz sentido fazer uma breve abordagem aos *browsers* nativos das plataformas móveis Symbian, Android, RIM, iOS e Windows Phone.

- **Symbian *Browser* (Sistema operativo Symbian)**

Em 2005, a *Nokia* criou um *browser* baseado na tecnologia Webkit para a série S60. Dependendo do dispositivo, são suportados os tipos de interacção *Focus*, *cursor*, *touch* e *multitouch*, (dos quais se falará posteriormente), sendo que muitos dos dispositivos suportam mais que um modo. Este é o *browser* base mais instalado no mercado, mas isto não significa que seja o mais utilizado (Firtman 2010).

- **Android *Browser* (Sistema operativo Android)**

Os dispositivos cujo sistema operativo é Android trazem consigo o seu próprio *browser* pré-instalado, também este, baseado em tecnologias Webkit. É muitas vezes comparado ao Safari no iOS devido aos *standards* e às funcionalidades que suportam. O Android *browser* suporta múltiplas janelas, *zoom* inteligente e muitas outras funcionalidades avançadas.

- **BlackBerry *Browser* (Sistema operativo BlackBerry)**

O *browser* BlackBerry tem sofrido uma série de melhorias com o intuito de proporcionar uma melhor experiência para os utilizadores dos dispositivos BlackBerry. Estas melhorias têm sido conseguidas devido à adopção do mecanismo Webkit. Esta tecnologia trouxe melhorias importantes tanto no desempenho como no suporte dos *standards* da *Web*. No recente sistema operativo, BlackBerry 6, o *browser* apresenta grandes melhorias, como o suporte de múltiplos separadores, assim como a selecção automática de rede que assegura que o *browser* está a utilizar a transferência de dados e ligação de rede mais rápida e menos dispendiosa (RIM 2010a).

- **Safari (iOS)**

Também o Safari é um *browser* baseado na tecnologia Webkit e vem pré-instalado com o iOS (anteriormente conhecido como iPhone OS). O *browser* é actualizado com novas funcionalidades que permitem criar uma melhor experiência de navegação. São suportadas funcionalidades que tornam possível a criação de animações

e transições simples e 3D baseadas em CSS (*Cascading Style Sheets*) e JavaScript. Apesar de não suportar Adobe Flash, possibilita experiências com HTML (*Hipertext Markup Language*), na sua versão 5, conseguindo-se animações semelhantes às possibilitadas através da ferramenta da Adobe. Este *browser* foi projectado para uma navegação baseada em *touch* e *multitouch*. Não são suportados os modos de interacção *Focus* ou *cursor* devido à falta de teclado físico nos dispositivos em que o iOS é instalado (Apple 2010).

- **Internet Explorer Mobile (Windows Phone 7)**

A Microsoft tem o seu próprio *browser* para dispositivos móveis. A versão Windows Mobile 6.5 utiliza o mesmo mecanismo que o *browser desktop*, proporcionando capacidades inteligentes de *zoom*. Antes do Windows Mobile 6.5 o *browser* suportava as funcionalidades de *Focus* nos *smartphones* e funcionalidades *stylus touch* (toque com caneta) para Pocket PC's. Agora, os dispositivos mais recentes no mercado suportam ambas as funcionalidades.

O novo Sistema Operativo, Windows Phone 7 é baseado na tecnologia *Silverlight*¹ (Microsoft 2010b) e tem uma interface totalmente *multitouch*. Este sistema vem com uma nova versão do Internet Explorer Mobile baseado no mecanismo do IE7 e com algumas funcionalidades do IE8 (Firtman 2010).

Browsers instalados pelo utilizador

Apesar de cada dispositivo com acesso à Internet ter um *browser* pré-instalado, existem inúmeros *browsers* livres ou pagos que o utilizador pode instalar após comprar o dispositivo. De entre todas as opções disponíveis salientam-se as seguintes:

- **Opera Mini & Opera Mobile**

Embora estes dois *softwares* pareçam semelhantes à primeira vista, as grandes diferenças residem na camada de suporte ao carregamento das páginas.

¹ *Silverlight* é uma plataforma de desenvolvimento que enriquece a experiência interactiva do utilizador na *Web* tanto em *desktops* como em dispositivos móveis (Microsoft 2010a).

No caso do Opera Mini, é utilizado o motor de renderização que se encontra no servidor Opera. Isto torna o acesso mais rápido, pois o carregamento das páginas é feito no servidor antes de ser enviada para o dispositivo móvel (Opera Software 2010). Contudo, isto significa que o servidor tem acesso ao conteúdo provocando uma falha na segurança. Este é um *browser* gratuito que funciona em praticamente todos os dispositivos, incluindo Android e iOS (iPhone, iPod e iPad). A partir da versão 4, são suportadas as funcionalidades de reprodução de vídeo, Ajax, leitura *offline* e *zoom* inteligente. A partir da sua versão 5, também são suportados múltiplos separadores e o modo de interacção *touch* em dispositivos que suportem esta tecnologia (Opera Software 2010).

Já o Opera Mobile, utiliza o mesmo motor de renderização que a sua versão para *desktop*, denominado Opera Presto. Este mecanismo é executado no dispositivo móvel utilizando o seu *hardware* para exibir as páginas *Web*. Este *browser* é também suportado por várias plataformas (Opera Software 2010). As versões mais recentes suportam *zoom* inteligente, *widgets*, Opera DragonFly (um conjunto de ferramentas para programadores) e Opera Turbo, um serviço que filtra e comprime os elementos multimédia das páginas *Web* nos servidores Opera, reduzindo o tráfego em 80% (Firtman 2010).

- **Firefox para dispositivos móveis (Fennec)**

Fennec é o nome do projecto realizado pela Mozilla para migração do *browser* Firefox para dispositivos móveis. A versão 1.0 do Firefox for *Mobile* é baseada no mesmo mecanismo de renderização do Firefox 3.6, denominado Gecko. Este mecanismo é *open source* e utilizado em muitas aplicações pela Mozilla. As suas características incluem a existência de múltiplos separadores, sistema de localização, possibilidade de sincronização com o computador utilizando o Firefox Sync (Mozilla 2010b). O suporte de *plug-ins* foi inicialmente desactivado por omissão, removendo a compatibilidade com alguns tipos de conteúdos populares na *Web*, como é o caso de Adobe Flash. Com o lançamento do Nokia N900, cujo Sistema Operativo, Maemo 5, é assente em Linux, em meados de 2009, o suporte de aplicações Flash já é possível (Nokia 2010).

- **Skyfire**

O Skyfire é um *browser* gratuito para as plataformas Android, Windows Mobile e Symbian S60 (Skyfire 2010a). Actualmente existe uma versão paga para iPhone que permite aos utilizadores deste dispositivo assistir vídeos em Flash, o que até então não era possível. Para que os utilizadores do iPhone consigam visualizar vídeos e animações em Flash, o Skyfire converte estes arquivos para HTML 5, suportado pelo iOS (Skyfire 2010b). Para além disto, o Skyfire suporta Ajax, bem como as principais tecnologias utilizadas na maioria das páginas *Web* actuais.

As páginas *Web* são pré-processadas no servidor Skyfire, utilizando o mecanismo de renderização Gecko, à semelhança do que acontece com a solução Fennec.

- **Bolt**

Bolt é um *browser* também gratuito e que permite ao utilizador ver páginas *Web* completas, incluindo conteúdos de vídeo e áudio. É baseado em Java ME, como o Opera Mini, sendo também compatível com dispositivos BlackBerry (Bitstream 2010).

- **Chrome**

Chromium é o nome de um projecto *open source* para o *browser* Chrome, para *desktop*. Sendo um projecto *open source* qualquer pessoa pode criar uma versão portátil para diferentes dispositivos móveis. Recentemente a Google criou uma extensão ao *browser* chamada *Chrome to Phone*. Esta extensão acrescenta um botão ao *browser* da Google que permite enviar informação do *browser* instalado no computador para um *smartphone* com o Sistema Operativo Android. Esta funcionalidade permite a transferência instantânea de vídeos, páginas que estejam a ser visualizadas, mapas, números de telefone, *links* ou texto seleccionado, que podem continuar a ser consultados, mais tarde, no telefone. No entanto só é compatível com a versão mais recente do Sistema Android, conhecida como Froyo ou Android 2.2 (Google 2010a).

2.1.2.2 Tipos de navegação

A experiência de navegar na Internet varia entre diferentes dispositivos e mesmo entre diferentes *browsers* no mesmo dispositivo devido às diferentes interfaces de utilização.

Uma página *Web* acessada através de um dispositivo móvel pode ser utilizada de diferentes formas. Cada *browser* permite um ou mais modos de interacção. São eles:

Focus – É o mecanismo mais frequente em dispositivos de gama baixa ou média. Os *smartphones*, por exemplo, que possuem teclas direccionais ou *trackball*, utilizam o *focus* como alternativa.

Com este mecanismo, é mostrada uma cor de fundo para o utilizador saber qual o elemento que está seleccionado. Em geral é utilizado em dispositivos que não são *touch*. Pressionando as teclas direccionais o utilizador altera o *focus* para o objecto (*link*, campo de texto ou botão) que deseja.



Figura 2 - Modo de interacção *focus* (retirado de Google)

Cursor – Neste tipo de interacção é emulado um cursor virtual que pode ser movimentado através das teclas direccionais, assim como o “clique”, através da tecla *Enter* ou “OK”. Para uma melhor experiência de navegação, muitos *browsers* fazem o salto do cursor directamente para a mais próxima para reduzir a distância entre as áreas “seleccionáveis”.

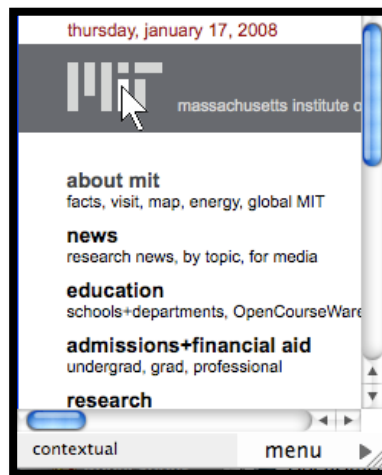


Figura 3 - Modo de interacção por cursor (retirado de Google)

Touch e Multitouch – A interacção é realizada através de gestos e movimentos rápidos de um, ou mais dedos no ecrã. O dispositivo interpreta-os como um comando. Há que ter em conta que, quando manipulamos os objectos com os dedos, a precisão é menor. A grande diferença entre *touch* e *multitouch* é que no primeiro caso o ponto de contacto é apenas um enquanto no segundo caso há a possibilidade de interpretar gestos constituídos por dois ou mais pontos de interacção.



Figura 5 - Modo de interacção *touch* (retirado de Google)



Figura 4 - Modo de interacção *multitouch* (retirado de Google)

A Tabela 2 apresenta algumas características relativamente a alguns *browsers* dos sistemas operativos falados anteriormente.

<i>Browser</i>	Mecanismo Webkit	<i>Zoom</i> inteligente	Modo de interacção
Symbian	Sim	Sim	Cursor ²
Android	Sim	Sim	Múltiplos ²
BlackBerry	Não (versões iguais e inferiores à 5.0)	Sim ²	Cursor
	Sim (Versão superior ou igual à 6.0)	Sim ²	<i>Touch</i> ²
Safari	Sim	Sim	<i>Multitouch</i>
Internet Explorer Mobile	Não	Não (Versões inferiores à 6.5)	<i>Focus</i>
		Sim (Versões superiores ou igual à 6.5)	<i>Touch</i> ²
Opera Mini	Não	Sim	Cursor <i>Touch</i> ²
Opera <i>Mobile</i>	Não	Sim	<i>Focus</i>
Firefox (<i>Fennec</i>)	Não (<i>Gecko</i>)	Sim	Múltiplos

Tabela 2 - Características dos *browsers* dos dispositivos móveis (adaptado de (Firtman 2010))

² Dependendo do dispositivo

2.1.2.3 Tipos de aplicações móveis

2.1.2.3.1 Aplicações Nativas

As aplicações nativas são desenvolvidas especificamente para as plataformas dos dispositivos em questão. Estas aplicações requerem adaptações e testes nos dispositivos em que são implantadas.

A grande maioria das aplicações nativas são certificadas, vendidas e distribuídas através de uma *application store* (“loja” online). Estas aplicações assentam sobre a plataforma do dispositivo podendo aceder a funcionalidades do mesmo, como por exemplo, ao sistema de localização, ao sistema de ficheiros e à câmara. Daí a necessidade de certificação antes da aplicação ser distribuída, para garantir que ninguém distribui uma aplicação insegura.

Exemplos de aplicações nativas:

Relativamente às aplicações nativas, algumas vêm instaladas de fábrica no dispositivo, outras podem ser compradas nas “lojas” online referente às respectivas plataformas.

A consulta da oferta é feita online e uma vez comprada ou descarregada a aplicação que se pretende, esta fica na memória interna do equipamento e pode ser utilizada quantas vezes o utilizador pretender. De seguida é feita uma pequena descrição das lojas online associadas aos Sistemas Operativos móveis e enumerada a lista das aplicações mais populares em cada um dos Sistemas.

A escolha é limitada pelo terminal e Sistema Operativo seleccionado, mas depois, existe um mundo de aplicações a que o utilizador pode ter acesso (Ferreira 2010).

Apresentam-se em seguida as principais lojas que se podem encontrar online.

- **App Store³, da Apple**

Esta “loja” da Apple oferece uma variedade de aplicações que complementam a oferta básica de funcionalidades do iPhone e iPod Touch. É rica em sugestões de utilidade, dando especial destaque aos jogos. De entre as categorias disponíveis salientam-se a educação, negócios, utilidades, finanças, livros, viagens, música, produtividade, entre outras. As aplicações mais populares, da loja portuguesa são: “Farmácias de Serviço”⁴ que mostra ao utilizador as farmácias que estão de serviço mais próximas da zona que escolher; o “Facebook”, uma rede social que torna mais fácil a ligação e partilha de informação com os amigos; e o “Skype” que permite efectuar e receber chamadas voip ou trocar mensagens instantâneas entre utilizadores do *software*.

De entre as aplicações pagas lideram, também na loja portuguesa: “Rádio Portugal”, um programa que “arruma” as rádios portuguesas no iPhone; “Mais Futebol” que fornece as últimas notícias relativas ao mundo do desporto; e a aplicação “Vamos Sair”, um guia de cidade com uma lista de eventos e respectivos locais na área da cultura e lazer.

- **BlackBerry App World⁵**

Apesar da RIM (2010b), ter uma posição marcante na área dos negócios, esta loja oferece uma variedade de aplicações para chegar a outro tipo de público. De entre os milhares de aplicações que se podem encontrar, uma grande parte corresponde a jogos.

A BlackBerry App World possui uma vasta gama de aplicações gratuitas. De entre as mais populares destacam-se: a aplicação “BlackBerry Messenger”⁶ que permite trocar mensagens instantâneas entre utilizadores que possuam também um equipamento BlackBerry; com o “Twitter” utilizador pode acompanhar amigos, partilhar interesses

³ <http://www.apple.com/iphone/apps-for-iphone> acedido em Setembro de 2010

⁴ <http://itunes.apple.com/pt/app/id389764591?mt=8> acedido em Setembro de 2010

⁵ <http://na.blackberry.com/eng/services/appworld/> acedido em Setembro de 2010

⁶ <http://appworld.blackberry.com/webstore/content/3729> acedido em Setembro de 2010

ou links em mensagens enviadas em tempo real; A aplicação “Pandora” é também muito popular na loja e consiste numa espécie de rádio personalizável em que o utilizador pode escolher o tipo de músicas que prefere ouvir.

No que concerne às aplicações pagas, destacam-se: “Hotel Tycoon Resort”, um jogo em que o utilizador é gerente de um hotel e tem que conseguir concretizar várias missões; a aplicação “StickyNote” que permite colocar as notas na página principal do telefone, e por fim a aplicação “One Touch Flashlight” que permite utilizar o *flash* da câmara do telefone como lanterna.

- **Android Market⁷**

A plataforma Android é suportada numa loja de aplicações que tem acompanhado a evolução da vasta variedade de dispositivos. Uma grande vantagem é o facto de mais de metade das aplicações disponíveis serem gratuitas.

Na vasta oferta que se encontra na loja, das aplicações pagas mais populares destacam-se: o jogo “Abduction! World Attack”, a ferramenta “MyBackUp Pro” que permite fazer cópias de segurança da informação guardada no equipamento e a aplicação “Power Manager” que permite aumentar a autonomia da bateria configurando algumas opções do telefone.

Das aplicações gratuitas as mais populares são: o jogo de cartas “Spades”⁸; “Places Directory” que permite navegar entre lugares próximos em categorias como: restaurantes, cinemas, hotéis, bancos, etc.; e a aplicação “Shazam”⁹ que permite identificar, armazenar e comprar músicas ou até saber mais sobre um artista.

- **Windows Phone MarketPlace¹⁰**

Uma das últimas a chegar ao mercado, a loja oficial de aplicações para *smartphones* com a versão móvel do Sistema Operativo da Microsoft, oferece neste momento um número limitado de aplicações, rondando a meia centena.

⁷ <http://www.android.com/market> acedido em Setembro de 2010

⁸ <http://www.bytesequencing.com/> acedido em Setembro de 2010

⁹ <http://www.shazam.com/> acedido em Setembro de 2010

¹⁰ <http://marketplace.windowsphone.com/> acedido em Setembro de 2010

O *site* da loja anuncia que as aplicações gratuitas mais populares no momento são: “Euromilhões Mobile”, a aplicação que permite registar chaves do jogo Euromilhões; “Microsoft My Phone” que permite sincronizar as informações do telemóvel com um *Website* protegido por palavra-chave onde se pode restaurar, partilhar e aceder às mesmas. A aplicação “Office Mobile 2010” é também das mais descarregadas. Esta aplicação é uma actualização gratuita para os telemóveis Windows Mobile 6.5 com a versão anterior do Office instalada.

No que diz respeito às aplicações pagas, as que se destacam são: “NDRIVE” que transforma o telefone num sistema de navegação, o jogo de corridas de carros “Ferrari GT Evolution” e a aplicação “SPB *Mobile Shell*” que apresenta uma nova interface de utilizador, mantendo as informações do dispositivo.

- **Symbian Horizon¹¹**

A Symbian Horizon funciona em moldes diferentes das outras “lojas”. Trata-se de uma página *Web* onde os utilizadores de sistemas compatíveis podem pesquisar aplicações e perceber em que lojas online estão disponíveis, já que se integra com as lojas próprias de vários fabricantes.

No universo das aplicações pagas dominam os jogos. É o tipo de aplicações que se encontram nos dez primeiros lugares das aplicações mais populares.

Relativamente às aplicações gratuitas, destacam-se como mais populares as aplicações: “Wikipedia”, que permite ler páginas da plataforma Wikipedia num formato adequado a ecrãs reduzidos; a aplicação “HandyWi” permite descobrir os *hotspots* disponíveis e a aplicação “Fring” que permite integrar vários serviços numa só aplicação, ou seja, é uma ferramenta de comunicação que permite efectuar chamadas de voz e vídeo, chat, etc. Tem integrados serviços como o Skype, Messenger, Google Talk, Facebook e Twitter.

¹¹ <http://horizon.symbian.org/> acedido em Setembro de 2010

2.1.2.3.2 Aplicações Web

Na secção anterior foram referidas algumas das aplicações nativas dos sistemas operativos mais populares. Neste ponto será tratada a forma de acesso à informação contida na rede, que proporcionou uma melhor experiência ao utilizador.

Como seria de esperar, um *mobile Website* é uma página *Web* concebida para dispositivos móveis. Este facto não deve ser confundido com a ideia de que um *mobile Website* é a exibição de uma página implementada para *desktop* em dispositivos móveis.

Os *Websites* implementados para dispositivos móveis são caracterizados pela sua simples apresentação de hiperligações que nos levam a outros níveis da página e pela sua natureza informativa, oferecendo poucos ou nenhuns elementos interactivos.

Apesar dos *mobile Websites*, baseados em WML (*Wireless Markup Language*) que não eram mais que uma série de *links*, serem relativamente simples de criar, não conseguiam mostrar o conteúdo de forma consistente em diferentes *browsers*. Com a introdução de *browsers* mais avançados em plataformas de dispositivos como o iPhone e Android, a qualidade das páginas *Web* para dispositivos móveis começou a melhorar drasticamente e com isso, aumentou a experiência de utilização. Por exemplo, em apenas um ano, o mercado de consumo da *mobile Web* dos Estados Unidos da América, deixou de estar nos cinco primeiros lugares para passar para o primeiro, em grande parte devido apenas à venda do iPhone (Firtman 2010).

Inicialmente os *mobile Websites* eram apenas para consumo de informação, actualmente as aplicações *mobile Web* permitem realizar qualquer operação, que em tempos apenas seria possível através de um *desktop*.

Uma aplicação *Web* pode definir-se como “um conjunto de programas que implementa um qualquer sistema de informação segundo o paradigma cliente-servidor suportado pelo protocolo HTTP (*HyperText Transfer Protocol*) e cuja camada interactiva está escrita em HTML de modo a que a interface com o utilizador seja assegurada pelos *browsers*” (Oliveira, Pereira et al. 2005). Usando XHTML (*eXtensible Hypertext Markup Language*), CSS e JavaScript, a sua execução em qualquer *Web browser* é possível, não havendo a necessidade de instalação no dispositivo. Permitem

que o utilizador interaja com o conteúdo em tempo real, realizando uma acção com apenas um “clique” ou um toque.

Com a criação de dispositivos com cada vez mais poder de processamento e a instalação de *browsers* igualmente poderosos, rapidamente começou a criação de aplicações *Web* optimizadas. Exemplo disso foram as aplicações criadas com o intuito de serem executadas apenas no iPhone. No entanto, a utilização da *Web* não explodiu apenas com os utilizadores deste dispositivo, pois as aplicações para ele desenvolvidas, funcionaram noutros dispositivos devido a terem sido criadas seguindo os padrões da *Web* (Web Standards Group 2010).

Exemplos de aplicações *Web mobile*

Dentro da vasta oferta de aplicações *Web*, existem já uma enorme quantidade de páginas *Web* adaptadas para dispositivos móveis, de forma a fornecerem ao utilizador uma experiência muito semelhante à da utilização de um *desktop*.

Para ter acesso a este tipo de aplicações é apenas necessário ter um dispositivo móvel que suporte acesso à Internet, um *browser* instalado e ligação à rede. Quando escrevemos o endereço electrónico no *browser*, em geral o servidor onde está alojado o *Website* detecta que tipo de *hardware* (telemóvel ou computador) e qual o *browser* que está a ser utilizado. O servidor faz esta verificação de modo a saber de que forma exibir o *Website*. Se detectar um dispositivo móvel, é mostrada a página que foi adaptada para ser exibida neste tipo de dispositivos.

As Figuras 6 e 7 são exemplos de *Websites* que foram adaptados para dispositivos móveis. À esquerda encontra-se o *Website* descarregado num *desktop* e à direita, o correspondente *mobile Website*. É de notar as diferenças entre os dois tipos de página. Este assunto será tratado mais adiante.

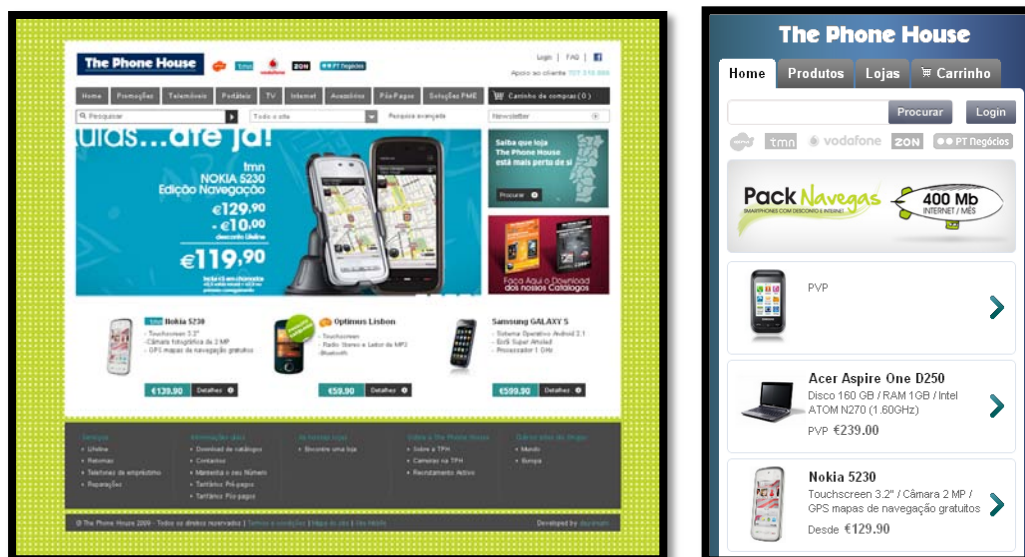


Figura 6 - Página Web "ThePhoneHouse", na sua versão *desktop*¹² (à esquerda) e versão *mobile*¹³ (à direita)

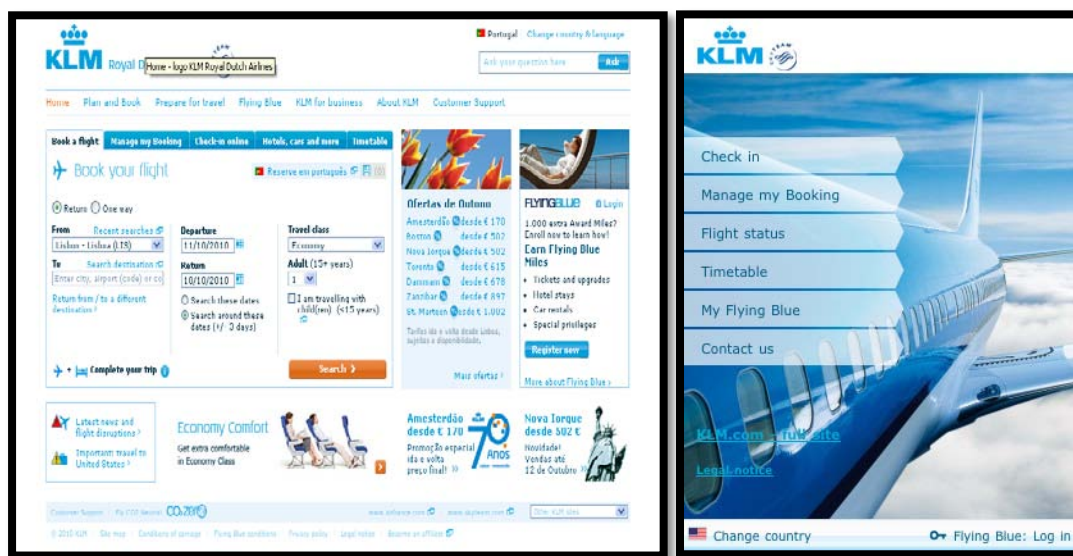


Figura 7 - Página Web "KLM", na sua versão *desktop*¹⁴ (à esquerda) e versão *mobile*¹⁵ (à direita)

¹² <http://www.phonehouse.pt/> acedido em Setembro de 2010

¹³ <http://m.phonehouse.pt/> acedido em Setembro de 2010

¹⁴ <http://www.klm.com> acedido em Setembro de 2010

¹⁵ <http://mobile.klm.com> acedido em Setembro de 2010

2.1.3 *Mobile 2.0*

2.1.3.1 *Web 2.0*

Quando se fala de *mobile Web 2.0*, não se pode deixar de introduzir o termo *Web 2.0*. A origem do conceito *Web 2.0* nasceu em meados de 2004 e está fortemente associado a Tim O'Reilly devido à conferência *O'Reilly Media Web 2.0* (O'Reilly 2005b).

Este fenómeno tecnológico e social tornou-se mais popular a partir das suas aplicações mais representativas: Wikipedia¹⁶, Youtube¹⁷, Flickr¹⁸, WordPress¹⁹, Blogger²⁰, MySpace²¹, Facebook²², OhMyNews²³ e a oferta das mais variadas ferramentas para captar utilizadores/geradores de conteúdos. Nesta nova versão da *Web* (*Web 2.0*) a Internet deixa de ser apenas uma forma de mostrar conteúdos tornando-se numa plataforma aberta construída tendo como base a participação dos utilizadores.

Assim, estes utilizadores actuam da forma que desejam: de forma tradicional e passiva, navegando apenas ou de forma activa criando e fornecendo conteúdos. O'Reilly citou a Wikipedia como “experiência radical e de confiança” (O'Reilly 2005b) onde qualquer utilizador pode definir um termo e qualquer outro o pode corrigir, transformando o utilizador num colaborador produtivo. No entanto, a Wikipedia é uma estrutura que apresenta um problema ao nível da participação dos utilizadores, pois está ao dispor de amadores e profissionais. Não existindo filtros sobre a entrada de conteúdos e portanto, a informação contida pode não ser a mais fiável, não servindo, por exemplo, como suporte de documentos científicos.

¹⁶ <http://www.wikipedia.org/> acedido em Outubro de 2010

¹⁷ <http://www.youtube.com/> acedido em Outubro de 2010

¹⁸ <http://www.flickr.com/> acedido em Outubro de 2010

¹⁹ <http://wordpress.com/> acedido em Outubro de 2010

²⁰ <http://www.blogger.com/> acedido em Outubro de 2010

²¹ <http://www.myspace.com/> acedido em Outubro de 2010

²² <http://www.facebook.com/> acedido em Outubro de 2010

²³ <http://www.ohmynews.com/> acedido em Outubro de 2010

O conteúdo que vemos, quando acedemos a uma página *Web*, sofreu um enorme impacto com a revolução da mesma, pois, o utilizador tem agora, a possibilidade de participar, gerando e organizando as informações ou até enriquecendo-o através de comentários, avaliação ou personalização.

Tim O'Reilly, numa breve definição, diz que a *Web 2.0* é a entrega de conteúdo continuamente actualizado em que as pessoas melhoram, utilizam e alteram os dados de múltiplas fontes, incluindo utilizadores individuais que permitem a alteração por outros, criando o efeito de rede através de uma arquitectura de participação (O'Reilly 2005a).

“Web 2.0 is definitely about people. I believe that the central principle of success in Web 2.0 applications is harnessing the collective intelligence of users” ... “In fact, the very essence of Web 2.0 is that we're making people part of the machine, and the machine part of the people, in new ways, blurring the boundaries between the two. Harnessing collective intelligence means that users are continually improving the application by their very interaction with it. Meanwhile, our communication, our knowledge acquisition, our social network, is increasingly computer mediated” (O'Reilly 2005a).

2.1.3.2 Aproveitamento da inteligência colectiva

Segundo Romaní e Kuklinski (2007), existem sete princípios subjacentes ao conceito *Web 2.0*, do qual é o principal promotor: a *World Wide Web* como plataforma de trabalho, o fortalecimento da inteligência colectiva, a gestão de bases de dados como competência básica, o fim do ciclo de actualizações de versões de *software*, modelos de programação leves na busca da simplicidade, o *software* não limitado apenas a um dispositivo e as experiências enriquecedoras dos utilizadores.

O diagrama da Figura 8 mostra que o aproveitamento da experiência colectiva é o centro da discussão relativa à definição de *Web 2.0*.

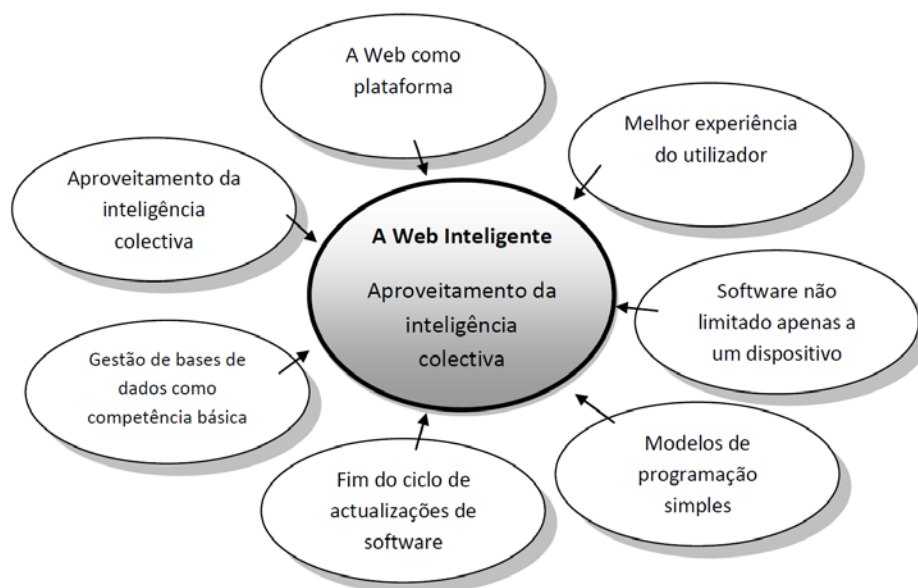


Figura 8 - Sete princípios que colocam a experiência colectiva no centro da *Web Inteligente* (adaptado de (Firtman 2010))

A inteligência atribuída à *Web* surge do colectivo, das pessoas. Assim, quando falamos da *Web* inteligente ou da inteligência colectiva estamos a falar do princípio “*Wisdom of Crowds*” - termo referente ao processo de levar em conta a opinião colectiva de um grupo de indivíduos ao invés de considerar apenas a opinião de uma pessoa (Surowiecki 2005).

De acordo com Maximiliano Firtman (2010), os sete princípios que colocam a experiência colectiva no centro da *Web Inteligente* resumem-se nos tópicos seguintes.

- **A *Web* como plataforma:** A *Web* é o único elo que nos une a todos como um conjunto. Assim, de maneira a aproveitar a inteligência colectiva e criar uma *Web* inteligente é necessário incluir o maior número de pessoas. A única forma de o fazer é tratar a *Web* como uma plataforma e criar padrões abertos.
- **Aproveitamento da inteligência colectiva:** Torna-se então o principal princípio ou o princípio número um.
- **Gestão de bases de dados como competência básica:** Por definição, para aproveitar a inteligência colectiva, devemos ter a capacidade de processar

grandes quantidades de dados. Portanto, os dados são a inteligência (Gatti e Jaokar 2009).

- Fim do ciclo de actualizações de *software*: Este ponto é válido para o *software* como um serviço e não como produto. O *software* como produto nunca pode estar actualizado devido à constante mudança de informação. O *software* como serviço mantém a informação relevante recorrendo ao máximo número de fontes possível.
- Modelos de programação simples: Utilizando modelos simples de programação é possível alcançar mais pessoas. Logo a possibilidade de trabalhar com muito mais fontes de informação.
- *Software* não limitado apenas a um dispositivo: Mais dispositivos para captura e melhor fluxo de informação entre eles, levam a um maior grau de inteligência colectiva.
- Melhor experiência do utilizador: Uma melhor experiência é necessária para permitir melhores aplicações *Web* levando a uma maior utilização e maior fluxo de informação.

2.1.3.3 O conceito *Mobile 2.0*

Há alguns anos, começaram a ser discutidas ideias acerca do conceito *Mobile 2.0*, baseado nos mesmos princípios da *Web 2.0*.

Embora a indústria dos dispositivos móveis tenha passado por muitas evoluções, os conceitos da *Web 2.0* são dos mais importantes, não só na tecnologia móvel, mas também na *Web* como um todo.

Normalmente pensa-se que “*Mobile 2.0*” consiste apenas em transportar a *Web 2.0*, até aos dispositivos móveis. Mas, nas várias pesquisas pelo termo “*Mobile 2.0*”, é difícil ou quase impossível obter uma definição simples e directa.

Ajit Jaokar – fundador da editora e empresa de pesquisa Futuretext²⁴ centradas na área das novas tecnologias da *Web* e *Mobile*, incluindo *Web 2.0* e *Mobile 2.0* – define

²⁴ www.futuretext.com acedido em Outubro d 2010

o termo *Mobile Web 2.0* como a extensão do conceito *Web 2.0* aplicado a dispositivos móveis.

“I see Web 2.0 as the Intelligent Web or ‘Harnessing Collective Intelligence.’ Mobile Web 2.0 extends the principle of ‘Harnessing Collective Intelligence’ to restricted devices.” (Jaokar 2006)

Numa tentativa de dar uma definição mais completa ao termo, Ajit Jaokar define no seu livro – *Mobile Web 2.0* – da seguinte forma:

- Agnóstica em relação às tecnologias: Qualquer tecnologia móvel pode ser utilizada para retirar inteligência a partir dispositivos móveis.
- Captura de inteligência no momento da inspiração: Os dispositivos móveis realizam bem esta tarefa na medida em que estão disponíveis no momento em que o utilizador tem uma ideia.
- Adição de metadados à informação partilhada pelo utilizador: Por exemplo, a inclusão de dados extra numa foto de forma a ser possível identificar a posição onde foi tirada.

Para Brian Fling (2009), o principal princípio do “*Mobile 2.0*” é reconhecer que estamos numa era de consumismo. A primeira tarefa do utilizador online é ler, para obter informação. No entanto, as pessoas que fazem parte do mercado actual não se vêm como meros consumidores, mas como criadores. Apesar de nos primeiros tempos da *Web*, serem necessárias ferramentas e *know-how* específico para poder publicar conteúdo na *Web*, com o início da evolução da *Web 2.0*, notou-se um aumento das ferramentas que permitiram publicar conteúdos com mais facilidade. Sendo as formas mais populares os *blogs*, redes sociais, *microblogging*, *media sharing* e *lifestreams*.

O *Mobile Web 2.0* é focado no utilizador como criador e consumidor de conteúdos “no momento de inspiração” sendo o dispositivo móvel o meio para aproveitar a inteligência colectiva que deriva desse momento (Gatti e Jaokar 2009). O utilizador tem então a possibilidade de capturar, gravar e partilhar conteúdo no momento sendo capaz de adicionar um novo contexto ao conteúdo (Fling 2009).

Para finalizar este tópico, realça-se a opinião de Daniel Appelquist (2006) – fundador da série de conferências *Mobile Monday London* e *Mobile 2.0*. Este autor afirma que o *Mobile 2.0* não é o futuro, mas sim, serviços que já existem ao nosso redor e que “amadurecem” a um ritmo impressionante. Estes serviços actuam em conjunto unindo a *Web 2.0* à plataforma móvel de forma a criar uma nova classe de serviços que potenciem a mobilidade mas que sejam tão fáceis de utilizar como é a *Web* actualmente.

2.1.4 Mobile Web design

Apesar dos *smartphones* poderem exibir praticamente todos os conteúdos *Web*, uma página formatada para ecrãs menores e com características que podem tirar vantagens de funcionalidades como o *touch screen* ou geolocalização, pode tornar a experiência de navegar na Internet através de dispositivos móveis, uma experiência melhor.

Quando se pensa em criar um *site* para ser exibido em dispositivos móveis há alguns aspectos que é necessário ter em conta. O primeiro será a decisão de, adaptar a página que já foi implementada para PC ou criar uma versão *mobile* completamente nova. Depois há que analisar qual o conteúdo que se deve manter e qual o que se deve eliminar.

Por outro lado, ajustar um *Website* para ser exibido em telemóveis pode ser um grande desafio. Os utilizadores que utilizam a Internet no telemóvel são muito menos pacientes que outros que utilizem o computador. Por isso, é muito importante ter em conta algumas considerações relevantes para atingir a satisfação máxima do utilizador.

Sendo assim, ajustar uma aplicação para dispositivos móveis significa repensar todo o propósito da aplicação e não apenas nas mudanças de visualização ou interacção. É necessário repensar o que é desejável e possível mobilizar para o ambiente móvel construindo e reconstruindo (Ballard 2007). Apresentam-se a seguir os principais factores a ter em conta.

2.1.4.1 Factores em conta no design para dispositivos móveis

- **Tamanho de ecrã**

Cada telemóvel tem um tamanho diferente de ecrã ou até mesmo diferentes posições, vertical e horizontal. Assim, o projecto de um *Website* móvel não pode ter uma largura fixa, mas sim um tamanho variável que se ajuste a diferentes tamanhos de ecrã. Utilizar percentagens, relativamente ao tamanho total do ecrã, pode ser uma opção. Outro método que pode ser utilizado é a limitação do *zoom* da página. Estes métodos podem ajudar à adaptação da interface a ecrãs de vários tamanhos. A resolução disponível nos diferentes dispositivos tem vindo a aumentar, por exemplo, a resolução 176x208 *pixels* era bastante comum há algum tempo, no entanto, agora vemos cada vez mais, dispositivos com ecrãs com tamanhos superiores. Por exemplo, o iPhone apresenta uma resolução de 320x480 e alguns *smartphones* apresentam já resoluções VGA (480x640 *pixels*).

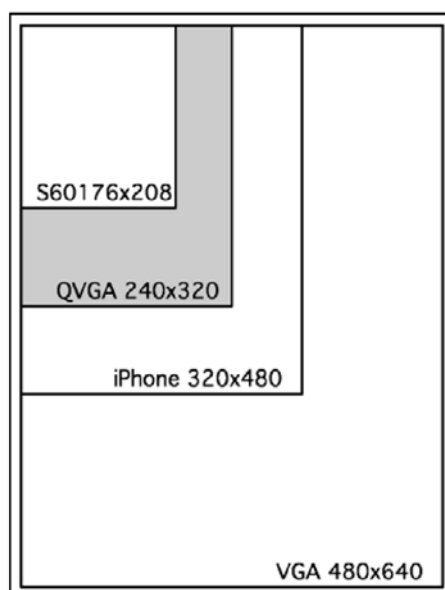


Figura 9 - Comparação entre os vários tamanhos de ecrã (adaptado de (Mehta 2008))

- **Layout**

O *layout* é um elemento muito importante da página, pois é através dele que o utilizador processa a página visualmente. Devendo este, permitir que a informação que o utilizador pretende encontrar seja de leitura rápida, simples e eficaz.

A definição de um *layout* é muitas vezes um problema, porque os dispositivos, como referido anteriormente, têm os mais variados tamanhos de ecrã, diferentes níveis de qualidade e resolução, alguns suportam *zoom* e diferentes tipos de interacção enquanto outros não.

Quanto mais informação se coloca num espaço razoavelmente pequeno e limitado, mais difícil se torna de ler e obter o que realmente se deseja, pelo que um *layout* com várias colunas não é admissível neste contexto. O espaço necessário para o conteúdo não deverá ir além do espaço físico do visor, caso isso aconteça é necessário recorrer às funcionalidades de *zoom* e dimensionamento da página, o que não é desejável.

Segundo Brian Fling (Fling 2009), devemos pensar no fluxo de informação como uma cascata, de cima para baixo, semelhante a um jornal. As informações contextuais tomam a posição do topo enquanto o conteúdo consome a maior parte do ecrã e os pontos de saída da página ficam no final. Desta forma consegue-se enfatizar toda a parte do conteúdo e minimizar o desejo do utilizador de sair da página.



Figura 10 - Fluxo típico da informação em páginas Web de desktop (adaptado de (Mehta 2008))

Os típicos *layouts* de páginas que vemos no *desktop* (Figura 10), não funcionam da melhor forma no telemóvel, já que não existe rato ou teclado no dispositivo móvel. É necessário escolher um *layout* que seja fluído verticalmente com blocos distintos com caminhos para diferentes elementos, pelo que a utilização de várias camadas de *links* pode não ser a melhor solução, podendo até torna-se demasiado cansativo.

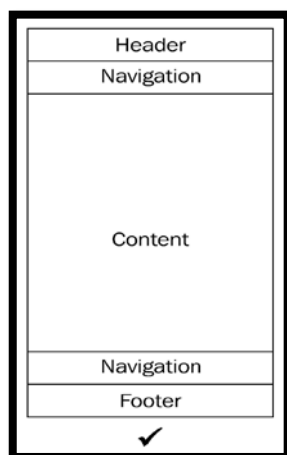


Figura 11 - Fluxo de informação desejado em páginas exibidas em dispositivos móveis (adaptado de (Mehta 2008))

Outro aspecto a considerar são as mudanças no *layout* quando a posição do dispositivo é alterada (de vertical para horizontal e vice-versa). O *layout* pode ser fixo: implementado com um determinado número de *pixels* de largura, ou fluído: capaz de se adaptar ao tamanho total do ecrã do dispositivo, independentemente da sua orientação.

- **Interacção**

Neste campo, é necessário ter em conta que o número de dispositivos móveis com interfaces *touch-screen*, tem vindo a aumentar significativamente. Isto significa que é necessário tornar as áreas seleccionáveis e os botões um pouco maiores e deixar mais espaço entre os *links*. *Links* ou botões que são pouco visíveis não devem ser usados, uma vez que ampliar uma área sempre que se pretende navegar para determinado sítio, nem sempre é desejável, em parte, devido à lentidão da funcionalidade *zoom*. Para melhorar estes problemas de interacção, muitos *sites* utilizam um *design* de navegação completamente independente, optimizado para *touch-screens* ou dispositivos que não utilizem nenhum dispositivo de entrada. Um bom menu será aquele que dá prioridade e ênfase às acções mais utilizadas pelos clientes da aplicação.

- **Conteúdo**

O conteúdo de uma página *Web* deve ser a prioridade máxima. É importante criar um *design* que enfatize aquilo que é mais importante para os seus visitantes. Com um

ecrã muito pequeno, a página deve ser o mais limpa possível, com poucas imagens e tendo o mais espaço possível em branco. Devido à simplicidade das páginas e a geral falta de opções, pode ser dado mais ênfase ao conteúdo centrando-o no utilizador.

Um exemplo em que vemos a simplicidade da implementação na versão *mobile*, é no *site* de notícias “The Onion”. Como é bastante comum, na versão *desktop* existem demasiadas informações, a maioria é publicidade, no entanto pode ver-se que na versão *mobile* esses *banners* não são visíveis, o que por si só torna o site muito mais simples, limpo e de fácil leitura. A Figura 12 mostra as duas versões deste *site*.

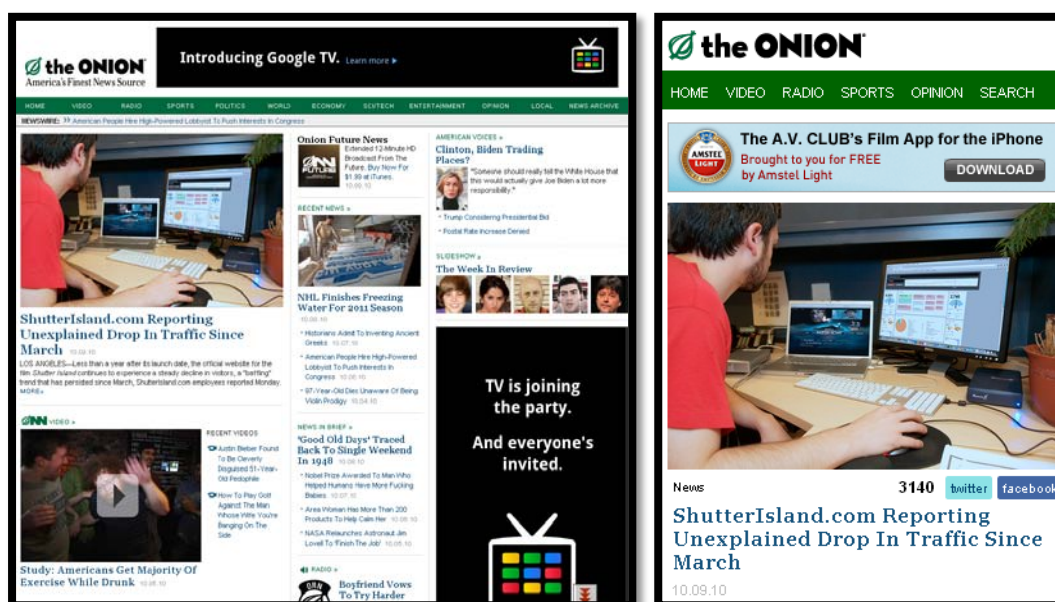


Figura 12 - Página Web "The Onion" versão *desktop*²⁵ (à esquerda) e versão *mobile*²⁶ (à direita)

²⁵ <http://www.theonion.com/> acedido em Outubro de 2010

²⁶ <http://mobile.theonion.com/> acedido em Outubro de 2010

- **Imagens**

Como foi dito anteriormente, o conteúdo da página é possivelmente o que mais interessa ao utilizador que a visita, logo será de máxima importância. Dado o tamanho reduzido do ecrã é necessário deixar o máximo espaço possível em branco, ou seja, com um número reduzido de imagens.

Como as ligações à Internet têm vindo a tornar-se mais rápidas e potentes, a utilização de imagens tem sido muito comum nos últimos tempos. Mas quando se trata de dispositivos móveis, o uso excessivo de imagens não é a melhor opção. O carregamento das imagens pode ser lento e a visibilidade do conteúdo pode ficar comprometida.

Sendo assim, é fundamental utilizar apenas as imagens que são necessárias para passar para o utilizador a mensagem que se pretende. Um logótipo e os ícones, na maioria dos casos, são suficientes para apresentar um *layout* simples e limpo.

- **Tamanho da página**

Como a capacidade da memória e o poder de processamento são limitados, muitos dispositivos impõem limites no tamanho total da página – incluindo CSS, XHTML MP e imagens. 10Kbytes é o mínimo que se pode esperar do tamanho da página e o máximo recomendado são 30Kbytes (Mehta 2008). O melhor será mesmo manter a página num tamanho menor possível. Para isso, reduzir na quantidade das imagens assim como a sua resolução, pode fazer muita diferença no tamanho da página *Web*. Muitos utilizadores desactivam a exibição de imagens de forma a economizar tráfego de dados, pelo que o melhor será colocar texto simples alternativo às imagens.

- **Links**

Como em *mobile* tudo é pequeno, devemos facilitar a percepção que os utilizadores têm da página. Os *links* não são excepção. É necessário torná-los mais fáceis de seleccionar. Isso pode ser feito recorrendo à utilização de mais espaço em branco ao seu redor e utilizar tamanhos de letra maiores. Quando um *link* for seleccionado, pode ser utilizada a mudança de cor ou a alteração da cor do fundo em vez de apenas utilizar o sublinhado habitual.

- **Cores**

Há uns anos atrás, os ecrãs eram apenas de duas cores, actualmente é possível utilizar nos dispositivos móveis um grande número de cores. Um problema que se encontra na utilização das cores é o número de bits utilizados para representar a cor de um *pixel* numa imagem. Diferentes dispositivos têm profundidades de cor (resolução tonal) diferentes, o que também pode alterar em muito a qualidade das imagens exibidas.

Quando são exibidas imagens muito complexas, devido à limitação de profundidade da cor dos dispositivos, pode ocorrer *posterization* – efeito que ocorre numa imagem devido à diminuição na profundidade da cor que provoca impacto visual na forma como as cores aparecem (McHugh 2010). A Figura 13 ilustra este efeito.

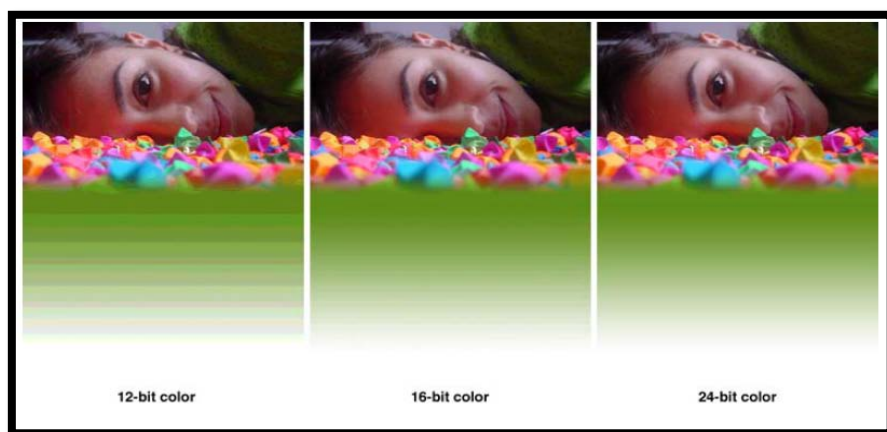


Figura 13 - Exemplo dos diferentes níveis de *posterization* que podem ocorrer em dispositivos com diferentes profundidades de cor (retirado de (Fling 2009))

Diferentes dispositivos suportam diferentes profundidades de cores assim como um número de cores limitado. A capacidade de exibir mais cores tem vindo a aumentar ao longo do tempo, com o aumento da capacidade dos dispositivos.

Na Tabela 3, pode ver-se o número de bits de profundidade, a quantas cores correspondem e alguns exemplos de dispositivos móveis com essas características.

Bits de profundidade	Número de cores que suporta	Descrição	Exemplos de dispositivos
12-bit	4,096 cores	Utilizado em dispositivos mais antigos. Podem ver-se algumas imperfeições em imagens.	Nokia 6800
16-bit	65,536 cores	Conhecido também como <i>HighColor</i> , ainda muito comum. Pode ainda causar imperfeições nas imagens.	HTC G1, BlackBerry Bold 9000, Nokia 6620
18-bit	262,144 cores	Podem ser vistas menos imperfeições. Oferece a visualização das cores quase perfeitas.	Samsung Alias, Sony Ericsson TM506
24-bit	17,6 milhões de cores	Também conhecido como TrueColor. Suportam milhões de cores e apresentam imagens nítidas.	iPhone, Palme Pré, Nokia N97

Tabela 3 - Cores suportadas por diferentes dispositivos móveis (adaptado de (Fling 2009))

As pessoas respondem às cores de maneira diferente. E neste contexto este facto é muito importante pois trata-se de um dispositivo bastante pessoal e que possivelmente é utilizado apenas por uma pessoa. Assim, utilizar as cores certas pode ser útil para transmitir a mensagem que se pretende e alcançar as expectativas dos utilizadores.

- **Tipo de letra**

O tipo de letra é outro aspecto muito importante quando se pensa no *design* de uma aplicação. Tradicionalmente, apenas existia um tipo de letra que se poderia utilizar. O único controlo que se tinha era sobre o seu tamanho.

Hoje, já existem mais opções, mas estes ainda são bastante limitados. Algumas plataformas têm tipos personalizados, assim como também podem variar de telemóvel para telemóvel. Portanto, a melhor opção pode passar por utilizar os tipos de letra que cada dispositivo tem por omissão ou *websafe fonts* – são as mais populares utilizadas

por *Web designers* para tornar o conteúdo dos seus *sites* mais legível para o utilizador (Gaillard 2010).

Existem dois grandes tipos de letra que podem ser utilizadas para tornar o design de um *site* mais atractivo: as fontes serifadas e as não-serifadas. As primeiras têm pequenos traços e prolongamentos no final dos caracteres.

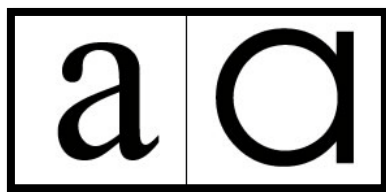


Figura 14 - Exemplos da letra "a" serifada e não-serifada (retirado de Google)

Discutir entre tipos serifados e não-serifados é apenas válido para textos impressos, para um ecrã de computador ou de um telemóvel, que possui uma resolução bem menor, e um sistema de visualização por *pixels*, as letras serifadas não são adequadas. A diferença de espessura no traço e os detalhes de serifas que auxiliam a leitura no texto impresso causam problemas e distorções nos ecrãs. O espaçamento inadequado, pesos irregulares entre os caracteres e entre os traços de um mesmo carácter, causam confusão e dificultam a leitura, principalmente em tamanhos de letra pequenos. Sendo assim, a recomendação é de utilização de letras sem serifa e principalmente aquelas desenhadas especialmente para a visualização em ecrãs, como a *arial*, a *verdana* e a *tahoma*, que têm espessura regular e espaçamento adequado para que as letras não encostem umas nas outras (Stolfi 2002).

A utilização de ecrãs baseados em subpixel – divisão de um pixel em vermelho, verde e azul (RGB) a um nível microscópico, permite um maior nível de suavidade nas fontes utilizadas. A adição de *subpixels* permite que o olho veja maiores variações de cinza, criando imagens mais nítidas dos caracteres. As plataformas Microsoft Windows Mobile utilizam a técnica do *subpixel* com a tecnologia *ClearType* que implica a utilização de mecanismos *anti-aliasing*. As Figuras 14 e 15 ilustram o benefício da utilização desta tecnologia.

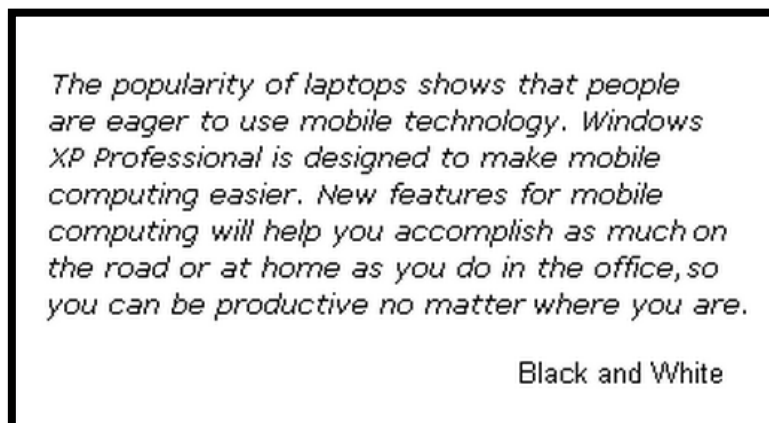


Figura 15 - *ClearType* inactivo (adaptado de (Microsoft 2010c))

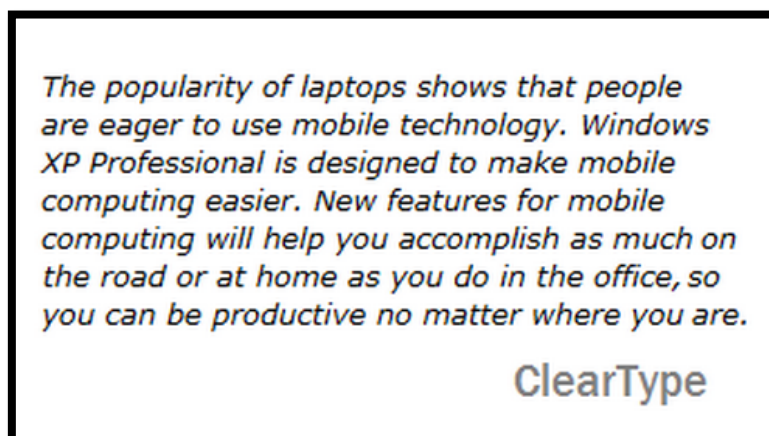


Figura 16 - *ClearType* activo (adaptado de (Microsoft 2010c))

Outra técnica consiste em utilizar uma grande densidade de *pixels*. A densidade de *pixels* é determinada dividindo a largura do ecrã em *pixels* pela largura do ecrã em polegadas. Com esta técnica aplicada a dispositivos móveis, quanto maior a densidade de *pixels* mais nítida é a imagem no ecrã. O texto é suavizado num ecrã com grande densidade de *pixels* ficando mais nítido também.

2.1.4.2 Simplificar

No contexto móvel, simplificar é fundamental. O *Website* móvel deve ser muito mais simples que o *Website* exibido no *desktop*. Eliminar elementos gráficos que não

são essenciais é uma maneira eficaz de otimizar a sua visualização num dispositivo móvel. Há que tentar encontrar opções de simplificar tanto o design como as funcionalidades do *site*. Isto pode significar refazer os menus, eliminar as imagens, quebrar o texto e dividi-lo em várias páginas e “re-trabalhar” tanto o *layout* como as suas funcionalidades (Snell 2009).

Devido à falta de espaço no ecrã e às ligações lentas à Internet, é importante que os utilizadores tenham acesso ao que é mais crucial para eles mesmos. É também essencial que o utilizador fique convicto de que o que é importante para si, está lá. As simples opções que quer tomar devem ser mais fáceis neste contexto que de qualquer outra maneira (Esposito 2010).

Um bom exemplo de simplicidade é a página Web <http://digg.com/> - uma rede social que reúne *links* para notícias, *podcasts* e vídeos enviados pelos próprios utilizadores. Combina *social bookmarks*, *blog* e *feed*. A página Web para dispositivos móveis é bastante mais simples que a exibida em PC, pois contém apenas alguns simples *headlines*, um *link* para efectuar *login* e poucas opções, básicas, de navegação.

“The best advice I can give you about mobile Web design is: Keep It Simple! However, that doesn’t mean Keep It Ugly.” (Firtman 2010)

2.1.4.3 Testar o design

Um dos grandes desafios na concepção de uma aplicação Web para dispositivos móveis é a grande variedade de dispositivos existentes e a forma como cada um exhibe a aplicação Web. Para começar, a simplicidade do *site*, como já foi referido, é muito importante. Nesta fase, também não o deixa de ser, uma vez que quanto mais simples, mais fácil será o seu teste, pois não há tanto por onde dar problemas.

Um enorme problema será testar a aplicação em todos os dispositivos e em todos os sistemas operativos, pois cada dispositivo tem as suas próprias peculiaridades que não podem ser ignoradas.

Neste tópico é mostrada uma lista de ferramentas gratuitas para testar a aplicação Web nos mais populares sistemas operativos referidos anteriormente.

Existem também ferramentas que verificam a aplicação e retornam relatórios detalhados sobre a eficácia nos dispositivos móveis.

Android Emulator²⁷ - O SDK do Android contém um emulador de dispositivos móveis. Um dispositivo virtual que é executado no computador. Mas, se o programador não necessitar do SDK e apenas quiser testar a sua aplicação, pode fazer o *download* do ficheiro executável que contém apenas o emulador sem ter que instalar o SDK. O emulador permite testar aplicações para dispositivos Android sem recorrer a um dispositivo físico. Este emulador simula todas as funcionalidades de hardware e software como um típico dispositivo.

MobiOne²⁸ - É uma ferramenta de desenvolvimento de aplicações que permite ao utilizador fazer rapidamente *mockups* e testar aplicações para iPhone e Palm Pre. Permite também aproveitar a experiência de uma comunidade que pode comentar e dar sugestões ao projecto. Esta aplicação permite programar, testar, corrigir erros, compilar e implementar aplicações *Web* para dispositivos móveis, sem que seja necessário conhecimento técnico aprofundado.

TestiPhone²⁹ - É um simulador online que permite rapidamente testar as aplicações *Web* para dispositivos iPhone. Esta ferramenta pode ser utilizada com os *browsers* Internet Explorer 7, Firefox 2 e Safari 3. As aplicações *Web* podem ser testadas segundo a orientação vertical e horizontal do simulador. É fundamental ter em mente que o iPhone não suporta *plugins* como Java ou Flash que os *browsers* de computador suportam e portanto, ao testar as aplicações neste simulador, podem ser exibidos conteúdos baseados nessas tecnologias.

BlackBerry Simulator³⁰ - Os simuladores BlackBerry permitem testar o *software*, o ecrã, teclado e a *trackwheel*. É possível também simular o comportamento do dispositivo em várias condições de redes sem fios. Quando esta ferramenta é

²⁷ <http://developer.android.com/guide/developing/tools/emulator.html> acedido em Outubro de 2010

²⁸ <http://www.genuitec.com/mobile/features.html> acedido em Outubro de 2010

²⁹ <http://www.testiphone.com/> acedido em Outubro de 2010

³⁰ <https://www.blackberry.com/Downloads/entry.do?code=060AD92489947D410D897474079C1477> acedido em Outubro de 2010

integrada com o BlackBerry JDE (Java Development Environment), os simuladores podem executar e encontrar erros de aplicações desenvolvidas no BlackBerry JDE.

Com o pacote BlackBerry *Email and MDS Services Simulator*³¹ é permitido simular alguns aspectos do BlackBerry *Enterprise Server*³² - ligação à rede para simular envio/recepção de emails, troca de mensagens instantâneas, tráfego de dados, entre outras.

WinWAP Smartphone Browser Emulator³³ - Permite utilizar os serviços WAP no computador. É um emulador do *browser* para dispositivos móveis que simula a forma como os serviços de Internet podem ser utilizados num *smartphone*. Este simulador permite realizar as mesmas acções no computador, como se estivesse a utilizar um *smartphone*. É permitido seleccionar os botões com o rato para fazer *scroll* e seleccionar *links*, e utilizar o teclado para introduzir texto. Podem também ser utilizadas *skins* que permitem alterar a aparência do telefone que aparece no ecrã do computador. Estas *skins* podem ser descarregadas por qualquer utilizador.

Windows Mobile 6.1.4 Emulator Images³⁴ - Este pacote adiciona imagens emuladas ao Visual Studio 2005 ou ao Visual Studio 2008 que permite testar aplicações para Windows Mobile 6.1.4 incluindo Internet Explorer Mobile 6. O simulador de imagens pode também ser utilizado de forma autónoma sem que seja necessário o *software* Visual Studio.

mobiReady³⁵ – A ferramenta mobiReady avalia o *Website* tendo em conta as boas práticas e padrões da indústria de aplicações *Web* para dispositivos móveis.

³¹http://docs.blackberry.com/en/developers/deliverables/16727/Installing_the_BB_Email_MDS_Serv_Sim_Pkg_816653_11.jsp acedido em Outubro de 2010

³² <http://br.blackberry.com/services/business/server/full/> acedido em Outubro de 2010

³³ http://www.winwap.com/desktop_applications/browser_emulator_skins acedido em Outubro de 2010

³⁴<http://www.microsoft.com/downloads/en/details.aspx?FamilyID=1A7A6B52-F89E-4354-84CE-5D19C204498A&displaylang=en> acedido em Outubro de 2010

³⁵ http://mobiready.com/launch.jsp?locale=en_EN acedido em Outubro de 2010

Depois de submeter o endereço online da aplicação, é fornecido um relatório com uma pontuação de 1 a 5 e uma análise aprofundada das páginas de forma a determinar quão bem o *Website* testado se irá comportar num dispositivo móvel.

Introduzindo a URL (*Uniform Resource Locator*), pode ser testada uma página individual, a linguagem *markup* ou até se pode testar um *Website* já existente.

W3C³⁶ mobileOK Checker³⁷ - É um serviço gratuito da W3C que possibilita verificar até que ponto a aplicação é “*mobile-friendly*”, ou seja se funcionará correctamente em dispositivos móveis. São executados vários testes ao *Website* e só se este passar em todos os testes é considerado “*mobileOK*”. Possibilita também testar conteúdo específico, como linguagens de marcação, *feeds* RSS, códigos CSS, entre outros. Tal como a *mobiReady*, é uma ferramenta online que disponibiliza um relatório detalhado com os erros encontrados.

iPadPeek³⁸ - Esta ferramenta online, permite visualizar como será exibida uma página *Web* num iPad, nas posições vertical e horizontal.

Opera Mini Simulator³⁹ - Permite a demonstração das funcionalidades do *browser* Opera para dispositivos móveis. Pode ser utilizado *online* em qualquer *browser* que suporte Java ou então descarregado e utilizado de forma independente.

2.2 Tecnologias

Actualmente, as aplicações *Web* móveis estão a ganhar cada vez mais espaço no mercado, relativamente às aplicações nativas, essencialmente por proporcionarem uma outra perspectiva ao nível de multi-utilizador. Desta forma, torna-se importante analisar as metodologias utilizadas.

³⁶ World Wide Web Consortium

³⁷ <http://validator.w3.org/mobile/> acedido em Outubro de 2010

³⁸ <http://ipadpeek.com/> acedido em Outubro de 2010

³⁹ <http://www.opera.com/mobile/demo/>

A decisão de desenvolver uma aplicação *Web* em detrimento de uma aplicação nativa, advém do facto das aplicações *Web* apresentarem as seguintes vantagens relativamente às aplicações nativas:

- São fáceis de desenvolver, usando HTML básico, CSS e conhecimentos em JavaScript
- São simples de implementar em vários dispositivos
- Permitem acesso simplificado – o acesso às aplicações *Web* é muito simples, basta escrever um endereço no *browser*
- Permitem o acesso imediato à versão mais recente, assim que são feitas actualizações
- Fornecem suporte para todas as plataformas – aplicação disponível a todos os utilizadores, independentemente do sistema operativo utilizado
- Não existem restrições de utilização e instalação – por exemplo em contexto empresarial, os utilizadores podem não ter permissões para instalar ou executar aplicações, restringindo a sua utilização
- Oferecem não só uma melhor experiência para o utilizador mas também um *design* mais rico, explorando os recursos do sistema do dispositivo
- O conteúdo é acessível em qualquer *Web browser* de um dispositivo móvel

Em contrapartida, as aplicações nativas apresentam as seguintes vantagens:

- Tornam a aplicação como o centro das atenções – os *browsers* bloqueiam muitas vezes e os utilizadores podem ter muitos separadores abertos, o que pode desviar as atenções da aplicação
- Permitem uma melhor utilização dos recursos do sistema – utilizar a aceleração disponibilizada pelo hardware faz com que as aplicações sejam bastante mais avançadas, tornando-se mais úteis e comportando-se de forma mais eficaz
- Permitem utilização *offline*, pesquisa por localização, acesso aos ficheiros do sistema, câmara, entre outros.

No entanto, dadas as necessidades de acesso simplificado, execução centralizada, suporte multi-plataforma, facilmente se verifica que a aplicação DSD-mobile se identifica como uma aplicação *Web*.

A diversidade de dispositivos móveis traz consigo uma variedade de plataformas e, junto com esta, uma gama enorme de ferramentas de desenvolvimento. No tópico seguinte serão apresentadas algumas ferramentas de apoio ao desenvolvimento deste tipo de aplicações.

2.2.1 Client-Side

2.2.1.1 HTML/XML/XHTML

HTML (*HyperText Markup Language*) é uma linguagem *markup* utilizada para produzir páginas *Web* que foi originalmente criada por Tim Berners-Lee (W3C 2010a). É uma linguagem descritiva baseada em *tags*, que não são mais que comandos de formatação da linguagem. Os documentos HTML são arquivos de texto simples que podem ser criados em qualquer editor de texto simples como o Bloco de notas ou TextEdit (Tittel e Noble 2008). Na realidade toda a base das páginas *Web* é o HTML que permite introduzir hiperligações, definir tamanhos e tipos de letras, intercalar texto com imagens entre outras funcionalidades, ou seja, o HTML nasceu da necessidade de reunir textos, gráficos e outros conteúdos multimédia em documentos electrónicos que podem ser enviados através da Internet (W3C 2010b).

O HTML deriva do SGML (*Standardized Generalized Markup Language*), uma metalinguagem muito complexa para ser usada como um instrumento geral para reforço do HTML. Em vez disso, o W3C criou um padrão conhecido como XML (*Extensible Markup Language*), baseado nas características mais simples do SGML. O XML é mais flexível, adequado para orientar o desenvolvimento de novas linguagens *markup* (Kennedy e Musciano 2002). Assim, surgiu o XHTML (*eXtensible Hypertext Markup Language*), uma versão mais limpa e rigorosa do HTML (W3Schools 2010c). XHTML é uma reformulação da linguagem HTML baseada em XML. É um passo adicional em direcção a uma versão mais flexível do *HyperText* com os benefícios da arquitectura XML e a integração de diferentes tecnologias (Standards 2010). A formulação original

do HTML possui algumas irregularidades que podem causar problemas para o *software* que decompõe o *markup*. Por outro lado, o XHTML utiliza uma sintaxe extremamente regular e previsível que é mais fácil de manusear (Tittel e Noble 2008).

No fundo, apesar de o HTML ser uma linguagem simples, o XHTML veio facilitar as coisas. Os documentos XHTML estão em conformidade com XML e como tal, são facilmente visualizados, editados e validados com as suas ferramentas padrão (W3C 2002).

2.2.1.2 CSS

Em 1997, surgiu uma das primeiras recomendações para a utilização de HTML, a separação da estrutura e da apresentação de uma página *Web* (Schafer 2010). Foi nessa expressa necessidade que surgiram as CSS (*Cascading Style Sheets*). Com a sua utilização, todo o código associado à apresentação está separado do código restante. Antes da existência das CSS's a manutenção das páginas tornava-se mais complexa, tornando, por sua vez, o código menos perceptível e mais desorganizado, podendo surgir redundância (Campos 2008).

Com a utilização das CSS é possível personalizar uma série de características das páginas *Web*, tais como: tipos de letra, cores, espaçamentos, dimensões, margens, entre muitas outras. A utilização desta metodologia traz grandes vantagens para os programadores:

- As páginas formatadas com recurso a CSS são mais fáceis de alterar
- A comunicação entre programadores é facilitada, pois todos seguem o mesmo padrão
- O controlo sobre as características das páginas *Web* é facilitado

Contudo, existe um grande problema na utilização das CSS, os diferentes *browsers* podem não interpretar da mesma forma um conjunto de instruções CSS, o que faz com que o resultado obtido possa ser diferente de *browser* para *browser*. Por esta razão, aconselha-se o teste das aplicações que utilizam CSS no maior número de *browsers* possível.

2.2.1.3 JavaScript

JavaScript é uma linguagem de programação que foi criada com o intuito de atender às necessidades de validar formulários do lado do cliente (*client-side*) e de interacção em páginas *Web* (MOZILLA 2010a). O código JavaScript é normalmente embutido em páginas HTML e é uma linguagem que permite a execução de scripts sem compilação preliminar (W3Schools 2010b), ou seja, através da interface DOM (*Document Object Model*) que representa e interage com objectos HTML, XHTML e CSS, os elementos podem ser tratados e manipulados dentro da sintaxe de linguagem em uso (W3C 2005).

A linguagem JavaScript permite não só colocar texto dinâmico numa página HTML como reagir a eventos, validar dados de um formulário, antes de este ser submetido ao servidor protegendo-o de processamento extra, detectar o *browser* utilizado, e dependendo do *browser* detectado, carregar uma página especificamente desenhada para o *browser* em questão ou até criar *cookies*, isto é, armazenar e recuperar informação introduzida pelo utilizador (W3Schools 2010b).

Esta linguagem fornece um conjunto de funcionalidades extra que permitem uma maior interactividade com as páginas. Contudo, é necessário ter em conta que os *browsers* executam JavaScript de formas distintas, o que poderá traduzir-se em problemas de compatibilidade. Por esta razão, as aplicações deverão ser testadas no maior número de *browsers* possível (Campos 2008).

2.2.1.4 Ajax

A utilização simples de HTML torna a navegação nas páginas *Web* cansativa e frustrante devido ao constante carregamento total das páginas a cada acção. No entanto, existem tecnologias que permitem ultrapassar esse problema, afectando-se, de forma assíncrona, apenas as zonas com as quais o utilizador acabou de interagir, não tendo de esperar que toda a página volte a ser carregada (Ullman e Dykes 2007).

AJAX (*Asynchronous JavaScript and XML*) é um método de implementação de páginas *Web* com o qual os pedidos do utilizador são processados de forma imediata. O

AJAX é uma combinação de várias tecnologias que os vários *browsers* existentes já suportam (Microsoft 2009b).

A principal vantagem da utilização de AJAX é a resposta imediata à acção do utilizador. Assim, este pode ter uma experiência assíncrona, ou seja, pode fazer várias tarefas ao mesmo tempo, não tendo que esperar que a página processe as informações para executar outra acção. Portanto, a cada novo pedido do utilizador evita-se a execução de uma nova página ou o recarregamento da mesma (Niederauer 2007).

2.2.2 Hosting

2.2.2.1 Apache

Apache HTTP Server é o servidor *Web* mais utilizado desde Abril de 1996 (The Apache Software Foundation 2010). Os motivos para essa popularidade são a sua performance, segurança e compatibilidade com diversas plataformas (Campos 2008).

O Apache utiliza o gestor de processos do sistema operativo em causa, sendo a recepção e processamento dos pedidos feita por processos distintos (Brown 2008), permitindo a recepção de uma grande quantidade de pedidos. Para além de ser independente da plataforma, o servidor Apache também suporta uma vasta gama de linguagens de programação, tais como, PHP, CGI, Java, C/C++, Perl, Python e JSP (Hiner 2000).

O Apache para além de ser um *software* gratuito, é também *open-source*, o que significa que o seu código-fonte pode ser alterado por qualquer utilizador, ou seja, qualquer pessoa pode dar a sua contribuição no desenvolvimento do *software*. É graças a esta característica que o Apache tem vindo a ser melhorado ao longo dos anos.

2.2.2.2 IIS

O IIS (*Internet Information Services*) da Microsoft surgiu após o Apache e está obviamente disponível apenas para sistemas Windows. Apesar desta limitação quanto ao sistema operativo, o IIS suporta uma vasta gama de linguagens de programação, tais

como VBScript, Visual Basic, Java, C/C++, Perl, Python, JSP, PHP, ASP e a Framework .NET (Volodarsky, Londer et al. 2008).

A gestão de processos é feita de forma idêntica ao Apache, ou seja, existem processos separados para a recepção e processamento dos pedidos, o que permite a aceitação de pedidos mesmo quando o IIS está tecnicamente desligado (Volodarsky, Londer et al. 2008). Para além disso, a funcionalidade de armazenamento automático de dados na cache (Microsoft 2010f) faz com que as respostas a pedidos sejam efectuadas de forma mais rápida.

Apesar de ser grátis, o IIS apresenta-se incorporado no sistema operativo sendo portanto necessário obter licença do sistema operativo para que se possa utilizar o servidor *Web*.

2.2.3 SGBD

Inicialmente, os dados ficavam dependentes dos programas que os geravam e manipulavam sendo necessário alterar todo o programa, para alterar a estrutura da informação. Por isto, surgiu a necessidade de guardar a informação de forma persistente para que o acesso e a gestão fossem feitos de forma mais flexível, dando origem ao aparecimento do SGBD (Sistema de Gestão de Bases de Dados).

Um SGBD é, então, uma ferramenta que fornece uma interface entre os dados que são armazenados fisicamente na base de dados e o utilizador desses dados (Damas 2007).

Os dados contidos nas BD's (Bases de Dados), estão organizados e armazenados de forma persistente, estando também estruturados de forma independente relativamente às aplicações que os utilizam. Desta forma pode garantir-se que os dados continuam a existir mesmo após terminar a aplicação que os gere, terminar a sessão do utilizador ou desligar o computador.

Segundo Pedro Cardoso (2009), apresentam-se de seguida algumas características dos SGBD's.

- A independência dos dados relativamente aos programas que os utilizam
- A redundância controlada para que o acesso seja mais rápido

- A integridade dos dados assegurada por procedimentos de validação
- A abstracção dos dados por parte do utilizador relativamente ao armazenamento
- O acesso simultâneo aos dados
- A facilidade de obtenção de informação actualizada

Ultimamente a *Web* tem-se tornado o meio preferido para transmissão e disseminação de informações, acesso a dados e comunicação pessoal (Santos e Santos 2010). Deste modo, inúmeras aplicações vêm a ser desenvolvidas neste ambiente, sendo portanto uma necessidade que estas aplicações sejam baseadas em SGBD's.

Actualmente uma grande parte dos dados existentes na *Web* são geridos por SGBD's, pois, a *Web* veio possibilitar a distribuição da informação a um número ilimitado de utilizadores (através da Internet).

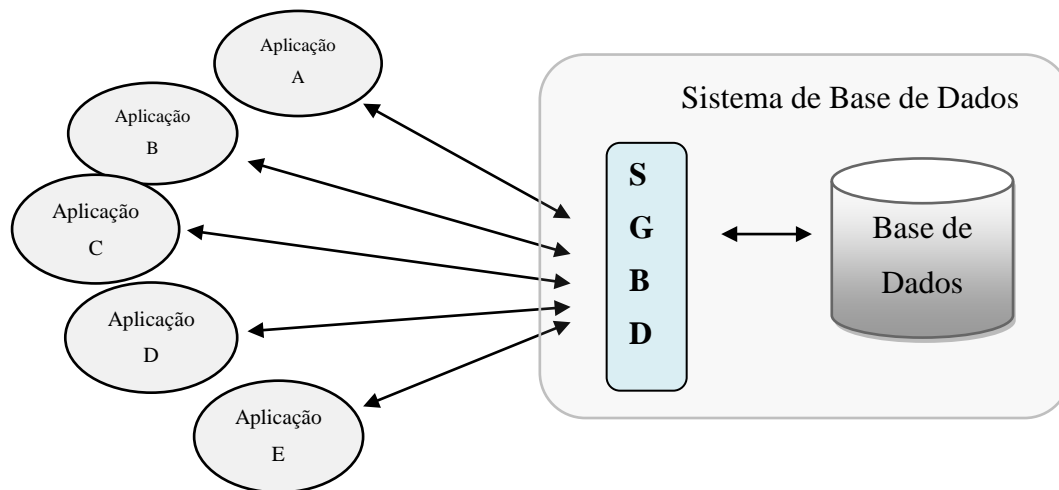


Figura 17 - Exemplo de um Sistema de Gestão de Bases de Dados (adaptado de (Cardoso 2009))

2.2.3.1 **SQLServer**

SQLServer é um SGBD apresentado pela Microsoft para o armazenamento e gestão de dados. Como todas as ferramentas da Microsoft, a portabilidade é o principal problema desta solução, uma vez que, apenas funciona em sistemas operativos Windows (Microsoft 2010e). Este SGBD pode ser utilizado através da maior parte das linguagens de programação, como Java e PHP, embora a integração com os produtos Microsoft seja feita de forma mais profunda, possibilitando a utilização de funcionalidades complexas de forma mais simplificada (Campos 2008).

Para garantir a segurança dos dados que gere, o SQLServer utiliza a funcionalidade de encriptação. A facilidade na sua manutenção, os relatórios de análise e o *backup* de dados de forma periódica, são também, algumas funcionalidades que este *software* suporta e que podem evitar eventuais situações indesejadas, como a perda de dados (Converse, Park *et al.* 2004).

Tal como as ferramentas proprietárias da Microsoft, referidas anteriormente, a aquisição deste produto não é grátis, sendo necessário o pagamento da licença de utilização. E, assim sendo, a aquisição de um sistema operativo Windows também é exigida.

2.2.3.2 **Oracle**

O Oracle foi um dos primeiros SGBD que surgiram no mercado. É bastante popular apesar do custo que lhe está associado. Uma clara vantagem deste *software* é a sua portabilidade, visto que pode ser utilizado em praticamente todos os sistemas operativos, como o Windows, Linux, Unix e Solaris (Oracle 2010c). E, assim sendo, a sua utilização também é suportada por várias linguagens de programação, havendo uma integração especial com o Java, dado que este se encontra embebido no Oracle (Oracle 2010c). Aconselha-se a utilização de um IDE Java, em especial destaque o Oracle JDeveloper, que é um IDE desenvolvido pela própria empresa que contém ferramentas que facilitam ainda mais a integração entre as tecnologias (Campos 2008).

Também existe uma versão gratuita do *software* Oracle, o Oracle Express, que tal como o SQLServer Express tem várias limitações, tanto a nível de dimensão das bases de dados como do próprio desempenho (Oracle 2010b).

2.2.3.3 MySQL

O MySQL é um SGBD grátis, direccionado para pequenas bases de dados. Este *software* caracteriza-se, não só, por ser *open-source*, mas também, por ser muito rápido, multi-tarefa e multi-utilizador (Oracle 2010a). Tal como o Oracle, o MySQL também pode ser utilizado na maior parte dos sistemas operativos, sendo a sua utilização suportada por uma vasta gama de linguagens. Também este SGBD suporta a replicação das bases de dados, embora de forma mais limitada nas suas funcionalidades (Corporation 2010). Esta replicação apenas é suportada num sentido, um servidor comporta-se como *master* e os *slaves* entram periodicamente em contacto com o *master* para obter os dados actualizados (Campos 2008).

Para a manutenção e gestão, existem muitas ferramentas visuais, que facilitam todo este processo. A destacar, o MySQL Workbench⁴⁰ que permite a construção de bases de dados com interfaces gráficas de simples utilização. Também as funcionalidades de *Backup* e *Restore* das bases de dados são suportadas.

Este *software* é gratuito, no entanto se o utilizador necessitar de versões com suporte e actualizações automáticas, é exigido o pagamento da licença de utilização.

2.2.4 Páginas Web dinâmicas

A base das páginas *Web* é o HTML, que suporta diversas funcionalidades como introduzir hiperligações, definir tamanhos dos tipos de letras, intercalar texto com imagens, entre muitas outras. No entanto, estas páginas são construídas de forma fixa, apresentando informação igual para todos os utilizadores. A este tipo de páginas *Web*, dá-se a designação de páginas *Web* estáticas (MacDonald 2006), onde a navegação é efectuada por documentos estáticos.

⁴⁰ <http://dev.mysql.com/workbench/>

Com a evolução no desenvolvimento de plataformas *Web*, surgindo a necessidade de funcionalidades que o HTML não suportava por si só, como por exemplo, a utilização de *scripts* para gerar páginas de forma automatizada. Esta automatização das páginas *Web* corresponde, por exemplo, a ter respostas diferentes perante contextos e/ou condições distintas, daí a designação de páginas *Web* dinâmicas (MacDonald 2006).

Existem dois tipos de páginas *Web* dinâmicas: de cliente e de servidor. Nas páginas dinâmicas de cliente todo o processamento é feito pelo *browser* do cliente, enquanto nas de servidor essa tarefa fica a cargo do servidor da página *Web*. Apesar de haver uma distinção deste tipo de páginas, estes podem ser utilizados em simultâneo. Com o surgir das páginas dinâmicas, a Internet deixou de ter um cariz meramente informativo, passando a ser um meio de comércio e publicidade, e como consequência, a complexidade das páginas aumentou de forma a responder às exigências dos utilizadores (Darie e Barnett 2008).

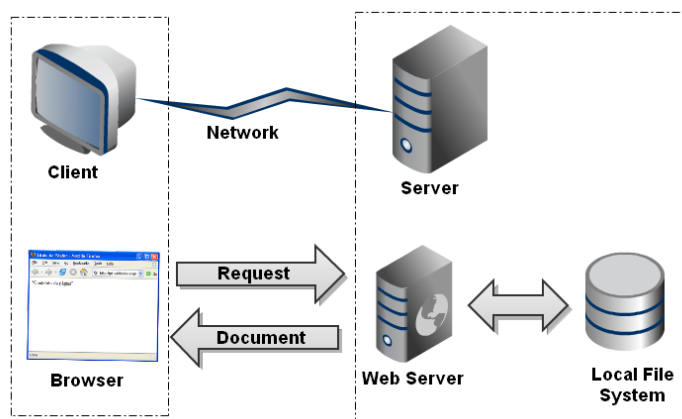


Figura 18 - Esquema ilustrativo de um pedido ao servidor de uma página Web (adaptado de (Darie e Barnett 2008))

2.2.5 Linguagens dinâmicas

2.2.5.1 PHP

PHP (PHP: *Hypertext Preprocessor*) é uma linguagem de código aberto, utilizada especialmente em conjunto com HTML, no desenvolvimento de aplicações *Web*.

O que distingue o PHP de outras linguagens, como por exemplo, Javascript, é que o código é executado do lado do servidor, gerando código HTML que é então enviado para o cliente. Desta forma o utilizador não tem conhecimento da linguagem utilizada (The PHP Group 2010).

O PHP é suportado por grande parte dos sistemas operativos, como o Windows, Linux e MAC OS (Converse, Park et al. 2004). Para além dos sistemas operativos, também grande parte dos servidores *Web* suportam esta linguagem, sendo até suportada a sua utilização no IIS.

O PHP tem um excelente desempenho, pois não há a necessidade de compilar código em tempo real, sendo apenas interpretado, o que requer menos recursos, sendo consequentemente mais rápido (Campos 2008).

2.2.5.2 JSP

JSP (*JavaServer Pages*) é uma tecnologia Java que permite o desenvolvimento de aplicações *Web* dinâmicas. Caracteriza-se essencialmente pela utilização de *servlets*, que de forma muito simplista, são trechos de código que adicionam funcionalidades ao servidor *Web* (Bergsten 2003).

Relativamente à sua portabilidade, esta tecnologia é suportada em todos os sistemas operativos. Nos servidores *Web*, também existe suporte para grande parte deles, existindo alguns especialmente desenvolvidos para esta linguagem (Wang 2010).

Existe uma vasta gama de IDEs (*Integrated Development Environment*), que podem ser utilizados no desenvolvimento de aplicações *Web*, na sua maioria são de

licença gratuita, destacando-se o Eclipse⁴¹ e o Netbeans⁴², duas ferramentas que proporcionam um ambiente de desenvolvimento integrado, por exemplo, com os servidores *Web* (Shrine 2004).

Depois do cliente fazer o pedido ao servidor, existem duas fases distintas: a tradução e o processamento do pedido. Na fase de tradução todos os elementos JSP são convertidos em código Java, que corresponde ao comportamento dinâmico pretendido. Após isso, o código java é compilado, gerando os ficheiros *class*, a partir dos quais é possível a execução na fase de processamento do pedido (Campos 2008).

2.2.5.3 ASP

ASP (*ActiveServer Pages*) é uma solução apresentada pela Microsoft com vista a facilitar e simplificar o desenvolvimento de páginas *Web*, tornando-as dinâmicas e interactivas (W3Schools 2010a). E por este motivo, a sua portabilidade é um dos principais problemas, pois as suas funcionalidades só estão disponíveis na totalidade em sistemas operativos Windows e em servidores IIS.

Uma das características do ASP.net é a independência da linguagem utilizada, isto é, suporta uma vasta gama de linguagens, como o C#, VB.net, J#, C++, etc. Esta independência de linguagem é uma grande vantagem da plataforma .NET, visto não ser necessária aprendizagem de novas linguagens para a utilizar (Darie e Barnett 2008).

Devido à existência de ficheiros distintos para a interface e para a lógica de funcionamento, o código torna-se mais organizado, sendo portanto, a manutenção das aplicações mais simples, não causando grandes dificuldades aquando a correcção de problemas ou no aumento de funcionalidades (Campos 2008).

2.2.6 MVC

MVC⁴³ (*Model-View-Controller*) é um padrão de arquitectura utilizado no processo de desenvolvimento de aplicações interactivas (Sun 2002). O MVC organiza uma aplicação interactiva em três módulos separados (BluePrints 2010):

⁴¹ <http://www.eclipse.org/>

⁴² <http://netbeans.org/>

Model: É a lógica de negócio da aplicação. Representa os dados da empresa e as regras comerciais que governam o acesso aos dados assim como a sua actualização.

View: este módulo concretiza o conteúdo do *model*, ou seja, acede aos dados através do *model* e especifica como devem ser apresentados. É responsabilidade do módulo *view* manter a consistência na apresentação quando se altera o *model*.

Controller: Providencia a “ligação” entre o *model* e o *view*. É responsável por processar as solicitações vindas do *browser*, procurando os dados no *model*, e passando-os para o módulo *view* com a finalidade de os apresentar ao utilizador, isto é, controla o despacho de pedidos e o controlo de fluxo de informação.

O MVC permite separar a parte de design (comportamento dos dados, apresentação e controlo), de tudo o resto, diminuindo a duplicação de código, centrando o controlo num módulo, tornando assim a aplicação mais facilmente modificável (Lindberg e Rydin 2002).

A grande vantagem da utilização do MVC é que ao alterar o *layout* ou ao criar uma nova interface, nada altera as regras de negócio do sistema ou do modelo computacional e é a mesma coisa com o modelo computacional, se alguma regra de negócio muda, não prejudica a visualização do utilizador. Trata-se de um maior controlo e manutenção dos produtos de software, resumindo-se em produtividade.

A aplicação de padrões MVC permite múltiplas visualizações sobre um modelo único. Vários pontos de vista podem ser implementados com o mesmo modelo, isto é, durante a execução múltiplos módulos de visualização podem ser acedidos ao mesmo tempo e dinamicamente. A separação efectuada permite também encontrar modelos que substituem os que estão a ser executados no momento, tanto os de visualização como de controlo (Deacon 2005).

Alguns autores (Stankovic e Simic 2010) encontram como desvantagens na utilização do MVC, a utilização de elementos distintos – *model*, *view* e *controller* – em menus e texto simples, pois aumentam a complexidade sem fornecerem mais

⁴³ <http://social.msdn.microsoft.com/Forums/pt-br/mvcpt/threads>

flexibilidade. A sua utilização pode provocar também, um número excessivo de actualizações – se o utilizador realizar demasiadas actualizações é necessário ignorar a notificação de alterações desnecessárias.

Quando o MVC é utilizado no desenvolvimento de aplicações para dispositivos móveis, devem ter-se em conta que a necessidade de múltiplas modalidades de entrada e saída leva a interfaces mais complexas. Portanto devem ser considerados os padrões disponíveis de desenvolvimento de software para construir interfaces de utilizador, a fim de utilizar uma estrutura mais eficiente que responda às necessidades (Deacon 2005).

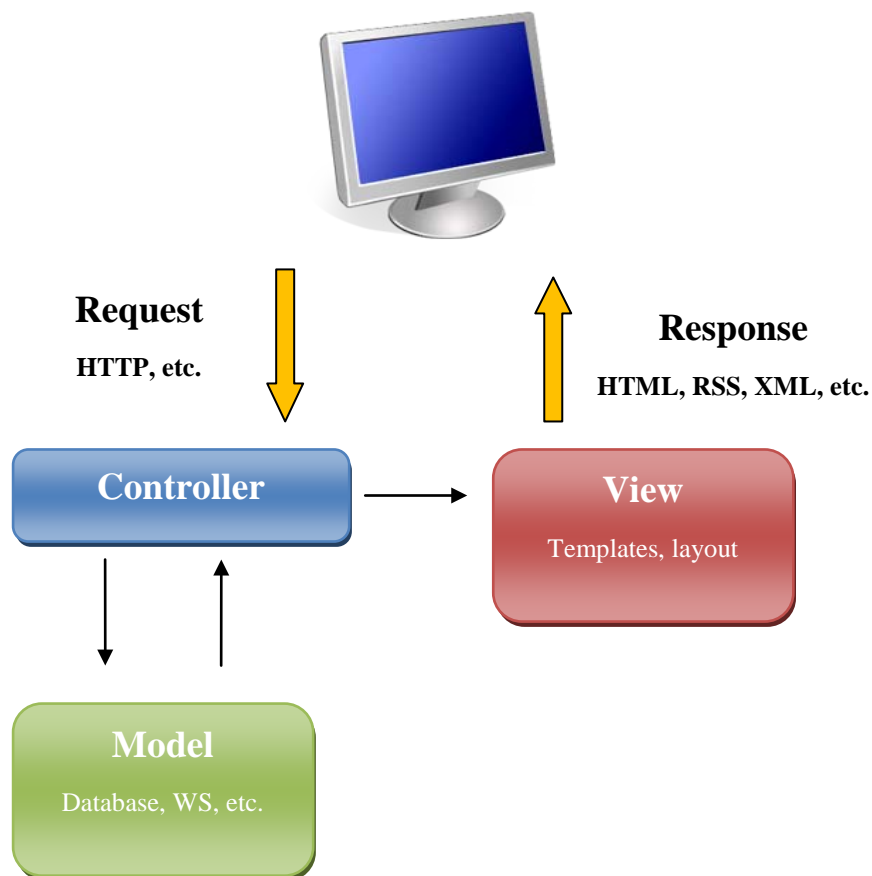


Figura 19 - Diagrama explicativo do padrão MVC (adaptado de (Microsoft 2010a))

2.2.7 Frameworks

2.2.7.1 .NET

O .NET é uma plataforma criada pela Microsoft com o objectivo da criação de sistemas e aplicações (Microsoft 2009c). O código gerado em .NET pode ser executado em qualquer dispositivo que possua a respectiva framework da Microsoft. Os programadores deixam de programar para um dispositivo específico e passam a programar para a plataforma .NET.

O .NET é composto por:

- CLR (*Common Language Runtime*) – fornece uma camada de abstracção sobre o sistema. Actualmente capaz de executar mais de 20 linguagens de programação diferentes interagindo entre si como se fossem uma só (Microsoft 2010d).
- *Base Class Libraries* – código pré existente para programação de tarefas de baixa exigência.
- Tecnologias e *frameworks* de desenvolvimento – soluções personalizáveis e reutilizáveis para programação em maior escala.

Para os dispositivos móveis, tem a vantagem da possibilidade de desenvolver aplicações para diferentes tipos de ecrã (Microsoft 2009a).

2.2.7.2 JQuery

O JQuery é um novo tipo de biblioteca do JavaScript. É uma biblioteca rápida e concisa que simplifica o processo de percorrer documentos HTML, de manipulação de eventos, de animações e interacções AJAX, de forma a obter um rápido desenvolvimento. Esta solução foi desenvolvida para alterar a forma como os programadores escrevem JavaScript (JQuery 2010). Esta *framework* permite que os programadores se concentrem mais na lógica do sistema e não nos problemas de compatibilidade entre os *browsers*. É um *software* livre cuja utilização é dirigida

segundo regras estabelecidas pelo MIT (*Massachusetts Institute of Technology*) e pelo GPL⁴⁴ (*General Public License*) (Silva 2010).

O JQuery destina-se a adicionar a uma página *Web*, interactividade e dinamismo, incrementando de forma progressiva e não obstrutiva a usabilidade, acessibilidade e o design de forma a enriquecer a experiência do utilizador. É utilizado nas páginas *Web* para: adicionar efeitos visuais e animações, procurar informações no servidor sem necessidade de recarregar a página, providenciar interactividade, alterar conteúdos, modificar a apresentação, simplificar tarefas específicas do JavaScript, entre outras (Silva 2010).

Na Tabela 5 mostra-se um exemplo de como o código JavaScript pode ser simplificado.

Sintaxe JavaScript	Sintaxe jQuery
<code>document.getElementsByTagName("p")</code>	<code>\$("p")</code>
<code>document.getElementById("um").setAttribute("class", "cor")</code>	<code>\$("#um").attr("class", "cor")</code>

Tabela 4 - Simplificação do código JavaScript utilizando sintaxe JQuery

2.2.7.3 SenchaTouch

SenchaTouch é uma *framework* para desenvolvimento de aplicações *Web* móveis baseadas em interacção *touch*.

A sua base tecnológica é o HTML5, CSS3 e JavaScript. Conta com uma extensa biblioteca de diversos efeitos de transições CSS e possui um completo controlo de eventos *touch*. A ideia principal do SenchaTouch é fornecer uma forma simples e flexível para os *developers* criarem as suas aplicações mais complexas. Com esta

⁴⁴ Designação da licença para *software* livre, idealizada por Richard Stallman no final da década de 1980, no âmbito do projecto GNU da Free Software Foundation (FSF).

framework é facilitada a criação de formulários, menus, efeitos de transições entre outros recursos de interface. Todos os componentes desta *framework* seguem padrões facilitando a comunicação entre outros componentes. Permite desenvolver aplicações *Web* que se assemelham, tanto na aparência como no funcionamento a aplicações nativas de dispositivos iPhone e Android (Sencha 2010).

2.2.7.4 Webapp.NET

Webapp.Net é uma *framework* de JavaScript que tira partido de tecnologias AJAX. É baseada em tecnologias *client-side*, e como tal, pode ser utilizada por qualquer tecnologia *server-side*, incluindo PHP, ASP.NET e Python. Não existe qualquer exigência relativamente à plataforma, isto é, pode ser utilizada em qualquer sistema operativo, incluindo MACOSX, Windows e Linux (Apers 2010).

Esta *framework* fornece um conjunto de API's⁴⁵ (*Application Programming Interface*) que economizam tempo e trabalho ao programador (Orenstein 2000).

As aplicações podem ser testadas num *browser* de *desktop* que utilize o mecanismo Webkit. Basicamente esta *framework* permite a “cópia” da interface dos dispositivos iPhone. Muito semelhante a esta *framework* é a iWebkit⁴⁶ que permite o mesmo tipo de desenvolvimento.

2.3 Avaliação da Usabilidade de Sistemas Móveis

A avaliação é uma das mais importantes partes do projecto de um sistema. É uma componente muito importante uma vez que permite medir, dimensionar e caracterizar o nível de sucesso de um sistema num determinado contexto. Para o melhorar, é fundamental avaliar para posteriormente corrigir os problemas, caso seja necessário.

A avaliação da usabilidade é uma necessidade para a viabilidade de um sistema, pois se a utilização deste for complexa, os utilizadores irão evitá-lo ou até trocá-lo por

⁴⁵ API é composta por uma série de funções acessíveis somente por programação, e que permitem utilizar características do software menos evidentes ao utilizador tradicional (Orenstein 2000).

⁴⁶ <http://iwebkit.net/> acedido em Outubro de 2010

um mais usável. A viabilidade de uma aplicação *Web* não é excepção sobretudo quando implementada para ser exibida em dispositivos móveis. As características inerentes à visualização e processamento de páginas *Web* em dispositivos com ecrãs reduzidos e processadores limitados fazem com que a avaliação de usabilidade seja um processo indispensável no desenvolvimento de um projecto.

A Usabilidade é um factor crítico para o sucesso de uma aplicação *Web* desenvolvida para ser exibida em dispositivos móveis (Bina, Karaiskos *et al.* 2007). No projecto de um sistema deste tipo é fundamental ter especial cuidado com o design, tendo especial atenção às restrições dos diferentes dispositivos, como o tamanho limitado do ecrã, as capacidades e controlos disponíveis para entrada de informação que são obrigados a ter menor qualidade de interacção quando comparados com um *desktop*.

Muitos estudos enfatizam a importância da realização de testes de usabilidade em contextos reais de utilização (Skov 2003), uma vez que diferentes condições de utilização abrangem vários tipos de comportamento. A navegação, por exemplo, pode ser dificultada devido às condições em que se está inserido devido a ambientes ruidosos ou sob diferentes condições de iluminação. Daí a necessidade de alargar a compreensão actual de usabilidade quando aplicada aos dispositivos móveis.

Durante o desenvolvimento de sistemas interactivos, a avaliação iterativa ao longo das etapas de projecto é fundamental no sentido de assegurar que o novo sistema tenha em conta as necessidades e limitações dos seus utilizadores, com vista à análise de requisitos (Dix, Finlay *et al.* 2004).

2.3.1 Requisitos

Pode-se dizer que um sistema é bem sucedido se responder de forma eficiente às necessidades dos seus utilizadores, tendo em conta o contexto em que se enquadra.

Numa definição superficial, um requisito é uma propriedade que deverá ser cumprida pelo sistema com o objectivo principal de resolver um problema do mundo real. Requisito de *software* é uma propriedade exibida por um *software* adaptado para a resolução de um problema em particular, resultando da combinação de requisitos de vários utilizadores inseridos num determinado contexto (Abran, Moore *et al.* 2004).

Os requisitos podem ser classificados em: Requisitos funcionais e não-funcionais. Os requisitos funcionais descrevem as funcionalidades que o sistema deverá oferecer, ou seja, funcionalidades que determinam o que o sistema irá fazer. Por exemplo, o sistema deverá enviar uma notificação por e-mail para o cliente quando uma encomenda é expedida. Os requisitos não-funcionais, também conhecidos como restrições ou exigências de qualidade, são expressos em termos de custos, requisitos de desempenho, confiança, segurança, fiabilidade, portabilidade, usabilidade, entre outros. São os requisitos não-funcionais que determinam a forma como os requisitos funcionais são levados a cabo (Maguire e Bevan 2002).

Quando se pretende desenvolver um sistema, tem-se como objectivo encontrar os requisitos de *software*, isto é, todas as propriedades a nível de funcionalidade que deverão ser incluídas nesse sistema. Para isso é necessário perceber as necessidades dos utilizadores e dos *stakeholders*.

2.3.2 Análise de requisitos

O sucesso dos sistemas começa com o entendimento das necessidades e requisitos dos utilizadores.

Como especificado na norma ISO 13407 (1999) o design centrado no utilizador começa com o completo conhecimento das suas necessidades. Os benefícios deste conhecimento podem incluir um aumento de produtividade, da qualidade de trabalho, uma redução nos custos de formação e suporte bem como um aumento da satisfação do utilizador (Maguire e Bevan 2002).

A base para a aplicação de diferentes métodos de análise de requisitos de utilizador é um processo simples, como é mostrado na Figura 20.

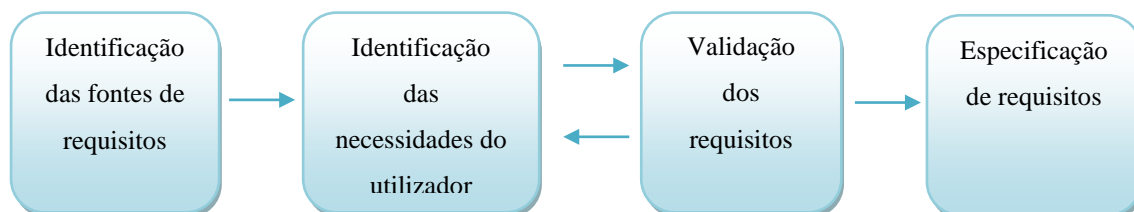


Figura 20 - Processo geral para a análise de requisitos (adaptado de (Maguire e Bevan 2002)))

A captura de requisitos está relacionada com a identificação das fontes de requisitos de *software* e a forma como os requisitos serão obtidos a partir dessas fontes. Esta é a primeira fase do processo onde serão identificados os *stakeholders* e estabelecidas as relações entre a equipa de desenvolvimento e os utilizadores/clientes (Sommerville 2007). Algumas dessas fontes são: os objectivos organizacionais, ambiente organizacional, domínio do problema, *stakeholders*, e o ambiente operacional (Goguen e Linde 1993). Existem alguns métodos para identificar as fontes descritas. Destacam-se:

- **Análise dos *stakeholders***, em que se identificam todos os utilizadores que podem ter influência ou ser afectados pelo sistema. Este método certifica que as necessidades de todos os utilizadores são tidas em conta.
- **Análise do contexto de utilização** do sistema. De modo a obter informação contextualizada, os utilizadores podem preencher um questionário onde são focadas as suas características, as suas tarefas e o ambiente operacional em que se inserem.

A Tabela 6 mostra os factores do contexto de utilização.

Grupo de utilizadores	Tarefas	Ambiente técnico
<ul style="list-style-type: none"> • Experiência e competências com o sistema • Conhecimento da tarefa • Treino • Qualificações • Competências linguísticas • Idade e género • Capacidades cognitivas e físicas • Atitude e motivação 	<ul style="list-style-type: none"> • Lista de tarefas • Objectivos • Saídas/Resultados • Passos • Frequência • Importância • Duração • Dependências 	<ul style="list-style-type: none"> • Hardware • Software • Rede • Materiais de referência • Outros equipamentos
Ambiente físico	Ambiente organizacional	
<ul style="list-style-type: none"> • Ambiente auditivo • Ambiente térmico • Ambiente visual • Espaço 	<ul style="list-style-type: none"> • Práticas de trabalho • Interrupções • Estrutura de Gestão e comunicação • Políticas de utilização de computadores 	

<ul style="list-style-type: none"> • Riscos de saúde • Roupa e equipamento de protecção 	<ul style="list-style-type: none"> • Objectivos organizacionais • Relações industriais • Características do trabalho
---	---

Tabela 5 - Factores de contexto de utilização (adaptado de (Maguire e Bevan 2002))

- A **análise de tarefas** envolve um estudo acerca do que o utilizador pretende fazer em termos de acções e/ou processos cognitivos para alcançar um objectivo. Pode ser efectuada uma análise detalhada de tarefas para se compreender o sistema, os fluxos de informação dentro dele, os problemas e as necessidades do utilizador. Uma das mais utilizadas é a análise de tarefas hierárquica onde as tarefas de nível mais elevado são decompostas em tarefas mais detalhadas e sequenciais (Dix, Finlay *et al.* 2004).
- **Observação** – corresponde a um observador verificar como os utilizadores desenvolvem as suas tarefas no contexto em que se inserem, tirando notas acerca das actividades que os utilizadores desempenham e do seu comportamento. A observação pode ser directa se for presenciada pelo observador durante a execução da tarefa. É indirecta quando as actividades dos utilizadores são gravadas e vistas posteriormente (Preece, Rogers *et al.* 1994).
- **Diário** – Manter um diário durante a execução das tarefas fornece um registo diário do comportamento do utilizador. Os diários permitem que o utilizador registe as actividades que estão envolvidas na utilização do sistema e podem levar à identificação das suas necessidades para um novo sistema ou produto.
- **Captura de vídeo** – regista o processo de utilização do sistema em contexto real. Pode ser visto posteriormente, tantas vezes quantas as necessárias. Permite visualizar comportamentos pormenorizados no contexto organizacional.

Depois de identificadas as fontes, pode então dar-se início à captura dos requisitos. Esta é uma das etapas mais difíceis do processo, principalmente devido à

aquisição dos requisitos do utilizador. Esta dificuldade prende-se muitas vezes com o facto do cliente/utilizador não saber exactamente o que deseja, ou transmiti-lo correctamente, tornando-se difícil compreender a descrição das tarefas pelos próprios utilizadores (Teixeira 2008).

A segunda parte do processo é então a identificação das necessidades dos utilizadores. Depois de estarem identificados é necessário perceber o que pretendem do sistema.

Para identificar estas necessidades existem também alguns métodos, dos quais são mais utilizados:

- **Inquéritos** – são constituídos por um conjunto de questões escritas e são colocados a uma amostra de utilizadores. São normalmente constituídos por perguntas de resposta “fechada” (opções de resposta dadas pelo inquiridor) e perguntas de resposta livre. Este método é normalmente usado para obter dados quantitativos assim como qualitativos, acerca dos problemas na execução de tarefas existentes ou no próprio sistema (Preece, Rogers *et al.* 1994).
- **Focus Group** – são discussões em grupo em que o objectivo geral é que cada participante estimule as ideias dos outros participantes presentes. Por um processo de discussão, o ponto de vista colectivo pode tornar-se valioso no desenvolvimento do produto (Bruseberg e McDonagh-Philp 2001).
- **Entrevistas** – utilizadores e *stakeholders* são questionados de forma a obter informação acerca das suas necessidades ou requisitos relacionados com o novo sistema. São normalmente estruturados com uma série de questões com hipótese de resposta extensa (Preece, Rogers *et al.* 1994).
- **Cenários e contextos de utilização** – Este método permite obter exemplos realistas e detalhados acerca de como os utilizadores executam as tarefas num contexto específico. O objectivo primário na construção do cenário é fornecer exemplos de uma futura utilização do sistema e

também ajudar a perceber e clarificar os requisitos dos utilizadores. Fornecem uma base para uma futura avaliação de usabilidade.

Segue-se então a etapa da validação dos requisitos. Depois de reunidos os requisitos iniciais, é necessário desenvolver uma arquitectura de informação de forma a mostrar esses mesmos requisitos. O *feedback* obtido por parte dos utilizadores perante esta arquitectura, permite validar ou refinar os requisitos.

Nesta etapa podem ser utilizadas, também, algumas técnicas. Das quais, as mais utilizadas são:

- ***Brainstorm*** – é uma técnica dinâmica, em que se explora a potencialidade criativa de um indivíduo ou de um grupo. Explora-se a criatividade da equipa, colocando-a ao serviço de objectivos pré-determinados. Nestas sessões são geradas novas ideias que permitem a criatividade dos intervenientes (Leffingwell e Widrig 2003).
- ***Card sorting*** - consiste num método de organização de informação de forma a garantir que a estrutura da mesma está adequada ao esperado pelo utilizador. Neste método existem duas técnicas que podem ser utilizadas: *Open Card Sorting* e *Closed Card Sorting*. O primeiro método consiste na organização da informação em grupos definidos pelos próprios utilizadores, o segundo na organização da informação em grupos pré-definidos (Usability.gov 2010a). Em suma, esta técnica permite identificar e delinear uma arquitectura de navegação de uma aplicação ou *Website*.
- ***Storyboards*** - São sequências de imagens que mostram a relação entre *inputs* do utilizador (acções do utilizador) e os *outputs* do sistema (resposta do sistema). Uma apresentação típica contém imagens que retratam características tais como menus, caixas de diálogo e janelas. As sequências de imagens fornecem uma plataforma para explorar e refinar as opções de requisitos dos utilizadores via uma representação estática do futuro sistema, mostrando-o a potenciais utilizadores (Trulock 2008).

- **Prototipagem** - técnica em que os *designers* criam em papel ou através de simulações, usando software apropriado, elementos da interface de utilizador (menus, botões, ícones, janelas, sequencias de diálogos, etc.) numa forma estática ou dinâmica. A prototipagem em papel é uma maneira fácil e rápida de criar esboços de páginas que podem ser usadas para fazer testes com utilizadores. Esta técnica é particularmente útil para recolher dados sobre conceitos e terminologia, navegação, conteúdos, *layout* e funcionalidade. As dificuldades e comentários dos utilizadores são registados por um observador. A prototipagem através de software fornece um maior nível de realismo em relação aos protótipos feitos em papel (Maguire e Bevan 2002).
- **Guidelines e standards** – São referidos por designers e especialistas em IHC para orientação em questões ergonómicas associadas ao sistema a desenvolver. A norma ISSO 9241 (1998b) abrange muitos aspectos de hardware, software e design de interfaces de utilizador, e contém um conjunto amplo de recomendações acerca de software ergonómico. Seguir guias de estilo, pode aumentar a coerência e reduzir o tempo de desenvolvimento (Nielsen 2000a).

Por último, tem-se a fase de especificação dos requisitos.

O **Mapeamento de tarefas e funções** especifica as funções do sistema que cada utilizador irá necessitar para as tarefas que vai executar. Este mapeamento mostrar a relação entre as funções e as respectivas exigências funcionais em forma de matriz, onde se podem fazer *trade-offs*⁴⁷ para adicionar ou para remover funções dependendo do seu valor para apoiar nas tarefas. É um método útil para sistemas multi-utilizador para garantir que as tarefas de cada tipo de utilizador são suportadas. Com o método de **categorização dos requisitos** especificação é feita por categorias, ou seja, os requisitos

⁴⁷ *Trade-off* é uma expressão utilizada numa situação em que existe conflito de escolha. Em termos económicos traduz o efeito de troca de um activo económico por outro que um determinado agente económico está disposto a efectuar (Infopédia 2003-2010).

são analisados separadamente tendo em conta os requisitos do utilizador, de usabilidade e empresariais. **Priorizar as necessidades dos utilizadores** é importante para que os recursos de desenvolvimento possam ser direccionados de forma adequada. O método de desenvolvimento DSDM (*Dynamic Systems Development Method*) é uma metodologia de desenvolvimento de software originalmente baseado em RAD⁴⁸ (*Rapid Application Development*). É uma metodologia de desenvolvimento iterativo e incremental que enfatiza o envolvimento constante do utilizador. O objectivo deste método é finalizar projectos no tempo e custo estimados através do controle e ajuste dos requisitos ao longo do desenvolvimento (Davis 1997).

2.3.3 Usabilidade

O conceito de usabilidade aparece normalmente associado a estudos de IHC e Ergonomia, referindo-se normalmente à facilidade com que uma interface de utilizador de um produto interactivo pode ser utilizada.

Deborah Mayhew (1999) afirma que usabilidade é uma característica mensurável de uma interface de utilizador apresentada num maior ou menor grau. Comportando duas dimensões fundamentais: a facilidade de aprendizagem e a facilidade de utilização.

Geralmente diz-se que um produto interactivo tem problemas de usabilidade quando os utilizadores encontram dificuldades na realização de tarefas utilizando a interface. Essas dificuldades podem levar à rejeição do produto por parte dos utilizadores. No fundo, o conceito de usabilidade está associado ao estudo da interacção onde, para além das propriedades da interface de utilizador, são também considerados outros atributos como a facilidade em aprender, facilidade de utilizar, eficiência da

⁴⁸ É um modelo de processo de desenvolvimento de *software* iterativo e incremental que enfatiza um ciclo de desenvolvimento extremamente curto, com alta qualidade e de baixo custo (Davis 1997).

utilização, minimização dos erros, satisfação dos utilizadores (Nielsen 1993) que deve ser alcançada de forma eficiente e eficaz (ISO 1998b).

Entende-se por eficácia a forma como o utilizador alcança os objectivos podendo ser avaliada em termos da finalização da tarefa com sucesso ou em termos da qualidade dos resultados obtidos. A eficiência refere-se à quantidade de esforço e de recursos necessários para se chegar a um determinado objectivo. E, para finalizar, entende-se por satisfação do utilizador, o nível de conforto que o utilizador sente quando utiliza uma interface na execução das tarefas.

Um sistema é socialmente aceite quando se encontra em conformidade com as normas sociais, culturais e morais dos seus utilizadores.

Existem várias técnicas para medir determinados atributos que dizem respeito à usabilidade de um sistema. Os critérios de medição de usabilidade estabelecidos pela norma ISO 9241-11 (1998b) são a análise de características requeridas do produto num contexto de utilização específico; a análise do processo de interacção entre o utilizador e o produto e a análise da eficiência (agilidade na execução das tarefas), da eficácia (garantia da obtenção de resultados esperados) e a satisfação dos utilizadores resultante da utilização do sistema.

Os testes de usabilidade são os mais comuns pelo facto da relação custo/benefício ser bastante aceitável. Consistem numa técnica formal que pode envolver utilizadores que representam o público alvo do sistema. Estes utilizadores são designados para desenvolver tarefas típicas e críticas, obtendo-se um conjunto de resultados para posterior análise. Esta captação de resultados pode ser através de gravação de vídeo ou por técnicas de observação, permitindo identificar os erros cometidos pelo utilizador, a rapidez com que se realizam as tarefas, se realizam a tarefa com sucesso ou não e se utilizam o caminho mais adequado. Para além disto é fundamental a observação do comportamento e expressões dos utilizadores, uma vez que podem revelar a sua satisfação e motivação.

Os utilizadores para a realização destes testes devem ser criteriosamente escolhidos tendo em conta os perfis dos vários utilizadores. É importante que os participantes no teste tenham um perfil similar aos potenciais utilizadores finais. É

igualmente importante que os participantes não estejam familiarizados com a aplicação, devem desconhecer os objectivos do observador e o propósito do estudo (Mitchell 2007).

O propósito das tarefas dos participantes é permitir que o observador examine os problemas que os utilizadores finais/clientes possam ter no contexto de utilização real. Através da utilização de uma lista de tarefas o observador pode verificar em pormenor em que partes da interface os participantes tiveram mais dificuldades e assim entender melhor como pensa o utilizador final. Com este conhecimento pode-se criar mais facilmente soluções para os problemas encontrados na interface de utilizador (Mitchell 2007).

Normalmente a este teste está associado um questionário que deverá ser respondido após a execução das tarefas propostas. Pretende-se com ele que os utilizadores manifestem a sua opinião sobre a aplicação. Do conjunto de questões que o questionário contempla, pretende-se conhecer, geralmente, a facilidade de utilização do sistema, consistência nos conteúdos, funcionalidades adequadas e perceptíveis, isto no que diz respeito à utilização geral do sistema. Existem ainda alguns aspectos mais específicos que se pretende conhecer, como por exemplo, o aspecto gráfico, a facilidade em navegar, mensagens apropriadas e esclarecedoras, tamanho dos caracteres, as cores, a quantidade de informação por página, etc.

Este tipo de testes pode ter o propósito de encontrar problemas de usabilidade de um produto ou com a finalidade de comparar dois ou mais produtos. Com o resultado destes testes pode-se melhorar as etapas seguintes do desenvolvimento.

A avaliação heurística trata-se de um método analítico que consiste na análise da interface de utilizador estruturado segundo um conjunto de princípios de usabilidade (Dix, Finlay et al. 2004).

Segundo Jakob Nielsen (2005), estes princípios de usabilidade podem ser encontrados ao longo da análise das dez heurísticas:

1. Visibilidade do estado do sistema;
2. Correspondência entre o sistema e o mundo real;
3. Controlo e liberdade do utilizador;

4. Consistência e standards;
5. Prevenção de erros;
6. Reconhecimento *versus* lembrança;
7. Flexibilidade e eficiência de utilização;
8. Estética e design minimalista;
9. Reconhecimento e recuperação de erros;
10. Ajuda e documentação.

Este tipo de avaliação produz um conjunto de problemas que correspondem à violação das heurísticas associada a uma escala de gravidade do problema. Nielsen propõe uma escala de 0 a 4, em que 4 corresponde a um problema muito grave e como tal de resolução urgente (Nielsen 1993). Os resultados são importantes para a equipa de desenvolvimento do sistema pois orienta-os na decisão de quais os problemas a serem resolvidos com mais ou menos urgência. A avaliação heurística é uma técnica bastante subjectiva e portanto é recomendável que seja realizada por elementos independentes do projecto. Segundo Nielsen cinco avaliadores é um número recomendável (Nielsen 1993; Nielsen 2005). Por um lado o aumento do número de avaliadores não levará ao conhecimento de muitos mais problemas de usabilidade, e por outro, o custo desta avaliação não valerá a pena quanto aos benefícios que trará com mais avaliadores (Sousa-Santos, Teixeira et al. 2007). Cada avaliador faz em primeiro lugar uma análise global, no sentido de compreender o sistema em geral. Depois deve fazer uma análise detalhada dos aspectos da interface tendo em conta as heurísticas de usabilidade. Os problemas encontrados devem ser anotados e atribuído um grau de gravidade de acordo com o entendimento que tem sobre a frequência, impacto e persistência do problema.

2.3.4 Soluções centradas no utilizador

Os interfaces gráficos foram desenvolvidos para dar controlo às pessoas sobre os seus computadores. As soluções centradas no utilizador tentam colmatar as necessidades dos utilizadores, adaptando a tecnologia às suas expectativas e nunca obrigá-los a usarem um interface que lhes coloca obstáculos desnecessários.

Este tipo de aplicações devem ser desenvolvidas envolvendo os utilizadores desde o início do projecto, fazendo com que o utilizador sinta que tem importância nas

decisões que são tomadas. A satisfação que o utilizador sente pode influenciar não só a sua motivação para aprender a utilizar a aplicação como a confiança na informação que a aplicação lhe fornece como ainda pode influenciar na percepção de facilidade de utilização.

“Por muito usável que possa ser uma aplicação, se o utilizador não estiver satisfeito com ela ou com a forma como 'não esteve envolvido' no processo, ele pode boicotar o seu uso e torná-la num fracasso.” (Gomes 2006).

Segundo Ivo Gomes (2006), o conhecimento do utilizador é também um factor fundamental para o desenvolvimento de soluções deste tipo. Definir os perfis dos diferentes utilizadores, o conhecimento da experiência relativamente a sistemas semelhantes, percepção dos termos utilizados na sua linguagem (pois por vezes uma simples palavra pode não ser perceptível num determinado publico alvo), são factores que permitem um melhor conhecimento do comportamento que pode esperar do público real. Faz parte deste processo a análise de tarefas que não é menos importante quando se pretende saber como se comportam os utilizadores perante o sistema. Os testes de usabilidade permitem também saber se o sistema corresponde às necessidades e expectativas do utilizador, identificadas anteriormente.

A maior parte das técnicas e metodologias anteriormente descritas, envolvem o recurso a utilizadores para desenvolver sistemas em que ele é o elemento central. *Focus group*, inquéritos/questionários, entrevistas, *card sorting*, prototipagem e testes de usabilidade são apenas alguns dos mais utilizados.

O desenvolvimento de soluções centradas no utilizador tem como principais vantagens: serem mais fáceis de aprender a utilizar, uma vez que o utilizador é envolvido no processo desde o início; os custos de formação são reduzidos, pela mesma razão anterior, o utilizador tem mais facilidade em aprender aquilo com que já entrou em contacto; a melhoria na usabilidade que conduz a uma maior satisfação dos utilizadores e vice-versa (Maguire 2001).

Na Figura 21 é mostrado o ciclo interactivo em processos de projecto centrado no utilizador.

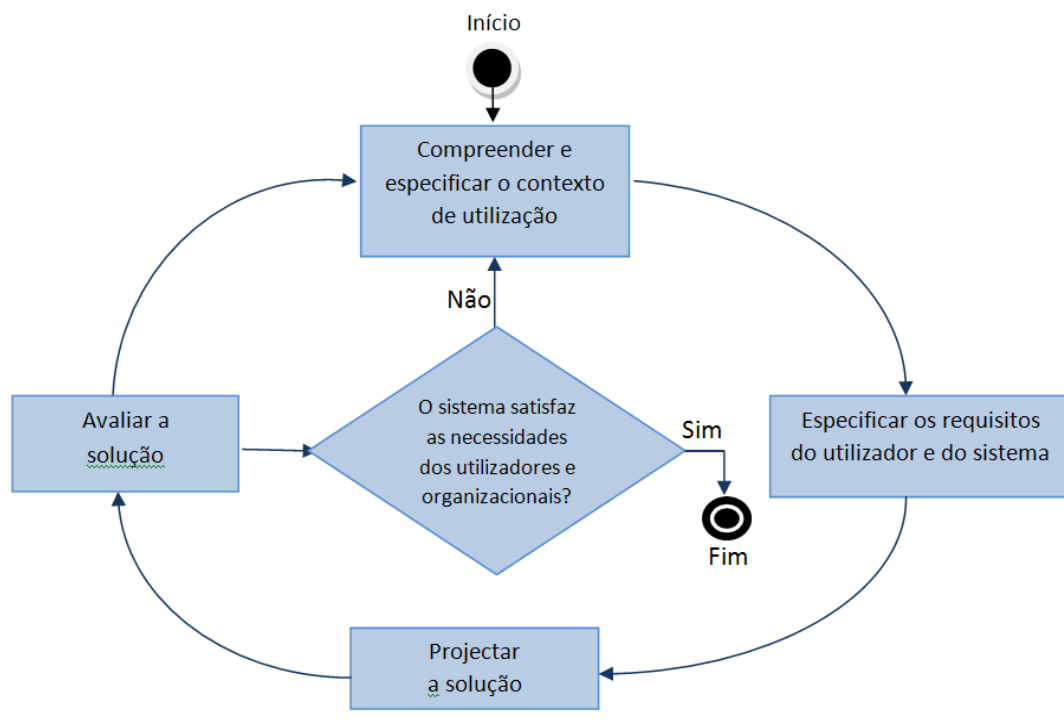


Figura 21 - Processo para o desenvolvimento de sistemas interactivos centrados no utilizador (adaptado de (Teixeira 2008))

Em cada uma das fases do processo, pode fazer-se uso de um conjunto de técnicas de recolha e representação de dados, desde questionário, entrevista, análise de documentação, observação directa, entre outras (Nuseibeh e Easterbrook 2000). A observação por ser um método em que se visualiza o utilizador no seu próprio ambiente de trabalho permite, mais facilmente, obter um conhecimento mais aprofundado das pessoas, tarefas e do contexto.

No fundo e em forma de resumo, o projecto de sistemas centrados no utilizador é uma abordagem para desenvolver produtos usáveis. É uma metodologia de desenvolvimento estruturada que envolve os utilizadores em todas as fases do processo a fim de criar produtos que respondam às necessidades dos utilizadores e organizacionais (Usability.gov 2010b).

2.3.5 Aplicação aos dispositivos móveis

Com o crescente avanço tecnológico na produção de dispositivos móveis, bem como nas tecnologias sem fios (*wireless*), estes sistemas começam a fazer parte da infraestrutura da tecnologia da informação das organizações.

Os dispositivos móveis, quando comparados com os computadores convencionais, ainda apresentam algumas limitações como capacidade de memória, poder de processamento, tamanho do ecrã e a bateria (Holtzblatt 2005). Em função dessas limitações, o desenvolvimento de software para dispositivos móveis torna-se uma tarefa desafiadora (Hosbond 2005).

Diante disto, verifica-se a grande importância de se facilitar a forma de utilização desses dispositivos móveis, isto é, disponibilizar uma interface bem estruturada, visando a facilidade de aprendizagem, simplicidade de uso e satisfação do utilizador ao interagir com o sistema.

Hoje os dispositivos móveis são cada vez mais comuns e mais necessários no dia-a-dia das pessoas. Os utilizadores destes aparelhos estão inseridos num contexto diferente dos utilizadores de *desktops*, além de apresentarem características específicas que devem ser consideradas no momento de projectar interfaces para os mesmos.

2.3.5.1 Usabilidade e os dispositivos móveis

Enquanto um *desktop* é utilizado para tarefas que exigem concentração do utilizador e são executadas durante um longo período de tempo, os dispositivos móveis são utilizados para aplicações mais rápidas e exclusivas do ambiente móvel. Para alguns autores (Lee, Schneider et al. 2005), a usabilidade de um dispositivo móvel depende de vários factores, incluindo o utilizador, o ambiente e as características do dispositivo.

a) Características do utilizador: A interacção do utilizador com um dispositivo móvel depende, até certo ponto, das suas características pessoais como “Flexibilidade e Destreza” (um utilizador adulto pode ter dedos maiores e precisar de um teclado maior) e “Conhecimento e Capacidade” (geralmente, os dispositivos mais úteis são os simples e mais intuitivos de usar, por outro lado, se o dispositivo for muito difícil de dominar, o utilizador não o achará útil).

b) Características do ambiente: O ambiente do utilizador, afecta a escolha do dispositivo. Em “Condições normais de funcionamento” um dispositivo móvel pode funcionar sob condições normais de trabalho do utilizador, devendo também trabalhar em “Condições extremas” (como calor, frio, humidade, luz natural e artificial).

c) Características do dispositivo: Os dispositivos móveis têm características próprias diferentes, que podem influenciar a usabilidade total. Características como “Tempo de inicialização” (em aplicações de tempo crítico pode requerer um dispositivo com inicialização imediata), “Integridade de dados” (se um utilizador não puder tolerar perda de dados e necessitar de armazenamento permanente, no próprio dispositivo), “Robustez/resistência” (os dispositivos móveis geralmente não são muito robustos e podem quebrar facilmente) e “Interface com o utilizador” (as características intrínsecas do dispositivo podem incapacitá-lo para desempenhar certas funções, devido à sua natureza de interface com o utilizador).

Sendo assim, a usabilidade dos dispositivos móveis não fica restrita à interface com o utilizador. As características do utilizador, do ambiente e dos dispositivos móveis são factores que influenciam a interacção e devem ser considerados tanto no processo de projecto de interface quanto na avaliação de usabilidade.

2.3.5.2 Contexto de utilização

Para analisarmos o processo de desenvolvimento de *software* para dispositivos móveis devemos entender o contexto em que eles são desenvolvidos. As aplicações móveis são projectadas e implementadas por razões de negócio, como melhorar a produtividade, aumentar a eficácia entre outras métricas. As aplicações móveis devem oferecer valor ou funcionalidade suficientemente grande para que o utilizador supere as restrições de usabilidade intrínsecas do dispositivo (Lee, Schneider et al. 2005).

Dada a importância da análise do ambiente em projectos de interface, Theo Mandel (1997), alerta que quando há alguma mudança física no ambiente, ou uma mudança total no conteúdo da informação que o indivíduo está a processar, então este muda imediatamente o foco de sua atenção para a nova informação. Isto pode ocorrer em virtude de uma variação luminosa, sonora, no movimento, nas cores ou complexidade da informação. O ambiente de utilização dos dispositivos móveis é

normalmente muito mais dinâmico em relação ao ambiente de utilização dos *desktops* e, portanto, requer a atenção do utilizador para outras tarefas além da interacção com o equipamento. O utilizador de um dispositivo móvel normalmente está envolvido em várias actividades que ocorrem simultaneamente, dividindo a sua atenção entre a utilização do equipamento e as outras actividades que está a realizar no ambiente que o rodeia.

Os dispositivos móveis estão ao alcance de uma grande variedade de utilizadores. Shneiderman (1998), de um modo geral, recomenda que os projectistas de interfaces de utilizador devem ter em consideração os diferentes tipos de utilizador quanto ao nível de conhecimento ou grau de experiência em sistemas de informação.

O utilizador casual é aquele que possui um dispositivo móvel, utiliza ocasionalmente algumas funções básicas, como por exemplo a agenda, calculadora e câmara digital, e o seu estilo de vida não está baseado na utilização de dispositivos móveis. Por outro lado, um utilizador frequente, leva sempre consigo o dispositivo móvel, fazendo uso de diferentes funções.

Sendo a usabilidade uma característica mensurável que indica o grau em que um sistema é fácil de ser utilizado, um alto nível de usabilidade na interface de utilizador dos dispositivos móveis, permite aos utilizadores executar tarefas rapidamente e com eficácia. O utilizador que tiver conhecimento e capacidade de utilização de sistemas computacionais poderá executar as tarefas de forma mais eficiente, no entanto, mesmo para os que não possuam experiência, as interfaces deverão ter um alto grau de usabilidade para que possam executar as suas tarefas, caso contrário, a aplicação estará comprometida. No entanto, nem sempre os utilizadores conseguem distinguir, quando estão descontentes com o produto, se o motivo é da interface, do dispositivo, da rede ou do serviço utilizado.

Tendo em conta estes elementos que cercam o contexto móvel, destaca-se a importância do projecto centrado no utilizador e, de acordo com a norma ISO 13407 (ISO 1999), compreender e especificar o contexto de utilização estabelecerá uma base sólida para as próximas fases de processo.

Em forma de conclusão, nesta secção verificou-se que a diversidade e as restrições intrínsecas dos dispositivos móveis têm aumentado significativamente a

dificuldade e a complexidade do projecto de interfaces de utilizador. Desse modo, a correcta compreensão do contexto de utilização, ou seja, identificar e compreender as características dos utilizadores e os seus diferentes níveis de conhecimento, o ambiente de utilização e as tarefas que serão executadas com o produto desde as fases iniciais do processo de desenvolvimento, é um factor determinante na usabilidade do produto final e fundamental para o sucesso de uma aplicação móvel.

3 Plataformas DSD

3.1 DSD Web – A plataforma existente

3.1.1 Objectivos da plataforma

A plataforma DSD, Distribuição de Serviço Docente foi idealizada para responder às necessidades de gestão de recursos humanos e físicos do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro (Campos 2008). As funcionalidades do DSD estendem-se à organização de horários, escolhas de dissertações e opções, inscrições em turmas, visualização de conteúdos, entre muitas outras. Com o crescente número de pessoas envolvidas (docente e discente), as limitações na realização destas tarefas começaram a aumentar. A necessidade de aumentar a eficácia e eficiência fez com que surgisse a plataforma DSD Web. Sendo o seu objectivo principal centralizar e facilitar a organização de grande parte das tarefas de uma forma eficiente.

A utilização da plataforma pode dividir-se em dois momentos: o primeiro diz respeito ao início de cada semestre em que os alunos assumem um papel mais activo, isto é, têm que realizar tarefas como as inscrições em opções, turmas e dissertações; o segundo momento corresponde ao resto do ano lectivo, em que os alunos assumem um papel menos activo e apenas utilizam a plataforma para consulta de conteúdo, como visualização de propostas de dissertações e opções, apresentações, horários, notícias, etc.

3.1.2 Funcionalidades *Front Office* (Aluno)

A plataforma DSD Web fornece um conjunto bastante vasto de funcionalidades. As entidades envolvidas na plataforma podem dividir-se em: **Docentes** – elementos participativos na submissão de dados; **alunos** – elementos participativos na inscrição em eventos; **coordenação pedagógica** – entidade responsável pela organização da distribuição de serviço docente e **Secretaria** – que auxilia a coordenação pedagógica na execução e organização de tarefas. No entanto, neste projecto dá-se ênfase às

funcionalidades que dizem respeito às tarefas dos alunos. Visto ser um projecto que se situa no ambiente dos dispositivos móveis, não se justifica a adaptação das tarefas de gestão de outros utilizadores da plataforma para estes dispositivos. Será visto mais à frente de que nestes casos, a adaptação das suas funcionalidades não lhes traria vantagens.

As funcionalidades *Front Office* consistem na parte da plataforma que trata da interacção directa com os utilizadores. Neste bloco *Front Office* existem os seguintes pacotes de gestão e as respectivas funcionalidades:

Pacote de Sessão: Iniciar Sessão, Terminar Sessão e Terminar Sessão por *time out*, estes gerem a entrada e saída do sistema, sendo os dois primeiros utilizados pelos actores Aluno e Docente e o último gerido pelo factor Tempo.

Pacote de Dissertação: Pesquisar Dissertação, Ver Dissertação, Listar Acordos, Ver Acordo e Assinar Acordo utilizados pelos actores Aluno e Docente, e correspondem à visualização das dissertações existentes, visualização de todos os dados associados a uma dissertação, ver a lista de acordos assinados ou não, ver detalhes de um acordo e assinar acordo relativamente a uma dissertação. Além destas funcionalidades, existem também várias utilizadas por apenas um actor. Correspondem aos alunos: Mostrar Interesse, Listar Dissertações Interessado, Ver Interesse, Remover Interesse e Editar Interesse que consistem nas funcionalidades que permitem ao aluno demonstrar interesse numa dissertação e remover ou alterar os dados dessa demonstração, ver a lista das dissertações às quais se mostrou interessado e os respectivos detalhes.

Pacote de Opção: Pesquisar Opção e Listar Colocações, utilizados pelos actores Aluno e Docente que permitem ver a lista de opções disponíveis e listar os alunos colocados em cada uma delas. Além dessas funcionalidades o Aluno possui as funcionalidades Adicionar Inscrição, Ver Inscrição e Editar Inscrição que correspondem ao processo de inscrição nas opções segundo a ordem desejada.

Pacote Horários: Visualizar os horários de todos os anos do curso e dos dois semestres dos anos lectivos 2008/2009, 2009/2010, 2010/2011.

Pacote Apresentações: Visualizar listagem de apresentações com as indicações do título da apresentação, data, hora e local onde irão decorrer. Fazer *download* da apresentação é também uma das funcionalidades existentes neste pacote.

Pacote Notícias: Visualização das últimas notícias de interesse.

Na Tabela 7 encontra-se uma breve descrição das funcionalidades existentes em cada pacote.

Pacote	Funcionalidade	Descrição
Sessão	Iniciar Sessão	O utilizador autentica-se na plataforma tendo acesso às funcionalidades que lhe dizem respeito.
	Terminar Sessão	Sair da plataforma.
	Terminar Sessão por <i>time out</i>	Passados 30 minutos de o utilizador ter iniciado a sessão, esta é automaticamente terminada, saindo da plataforma.
Dissertação	Editar Conta	Editar os dados relativos à conta pessoal.
	Pesquisar Dissertação	Pesquisar por uma Dissertação tendo ao dispor dois tipos de pesquisa: básica e avançada.
	Ver Dissertação	Ver todos os dados relativos a uma Dissertação.
	Ver Acordo	Ver todos os dados associados a um acordo.
	Assinar acordo	Assinar o acordo de uma Dissertação.
	Remover assinatura	Remover assinatura de um acordo.

	Mostrar Interesse	Mostrar interesse numa dissertação.
	Listar Dissertações Interessado	Listar todas as dissertações em que mostrou interesse.
	Remover Interesse	Remover o interesse mostrado por uma dissertação.
	Editar Interesse	Editar interesse mostrado por uma dissertação.
Opções	Pesquisar Opção	Pesquisar por uma opção tendo ao dispor dois tipos de pesquisa: básica e avançada.
	Listar Colocações	Ver a lista de colocações nas opções (provisória e definitiva).
	Adicionar Inscrição	Adicionar inscrição nas opções.
	Ver Inscrição	Ver inscrição previamente adicionada.
	Editar Inscrição	Editar a inscrição previamente adicionada.
Horários	Ver Horários	Visualização dos horários por ano e semestre.
Apresentações	Ver Apresentações	Visualizar listagem e documentos das apresentações.
Notícias	Ver Notícias	Visualizar últimas notícias.

Tabela 6 - Funcionalidades *Front Office* (adaptado de (Mota 2009))

3.1.3 Tecnologias

A plataforma actualmente existente apresenta a arquitectura apresentada na Figura 22.

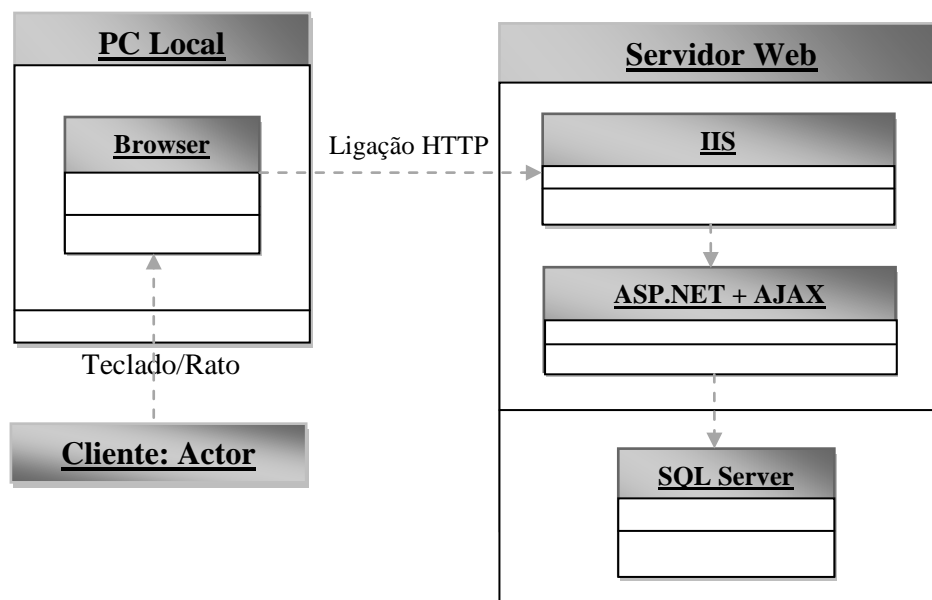


Figura 22 - Arquitectura da plataforma DSD (adaptado de (Mota 2009))

Esta arquitectura apresenta como Sistema de Gestão de Base de Dados (SGBD) o SQLServer 2005. Foi utilizado este sistema porque garante uma escalabilidade e desempenho adequados a um sistema de dimensões elevadas. Da mesma forma, se a sua utilização se estenda a mais departamentos não há problemas de adaptação para uma utilização mais intensa (Campos 2008). De forma a obter uma plataforma com páginas *Web* dinâmicas foi utilizada a tecnologia ASP.NET como linguagem base da plataforma. Embora o seu desempenho perca um pouco relativamente a outras tecnologias, o ASP.NET tira proveito de outras características que o tornam bastante competente (Campos 2008). Foi utilizada a tecnologia AJAX para dinamização da mesma (Mota 2009).

Como servidor, foi utilizado o IIS. Estudos anteriores verificaram que tanto o IIS como o Apache são extremamente competentes, mas a opção foi pelo primeiro devido

ao Apache ter a necessidade de módulos extra para funcionar com as mais variadas linguagens e sistemas (Campos 2008).

3.1.4 Conclusões

Neste capítulo, a plataforma *DSD Web* foi descrita de uma forma sucinta. Foram apresentados os seus objectivos principais, as suas funcionalidades *Front Office* para os utilizadores com o estatuto de aluno na Universidade de Aveiro. Foram abordadas também, as tecnologias utilizadas na sua construção e manutenção.

3.2 DSD-mobile

3.2.1 Contexto do projecto

Como foi referido anteriormente, tem-se notado uma crescente taxa de utilização de dispositivos móveis, cada vez mais avançados a nível de tecnologias que suportam, processamento, memória, etc. Numa Universidade esse crescimento na taxa de utilização é ainda mais evidente devido não só à faixa etária predominante mas também à forte componente tecnológica que o meio académico envolve.

O projecto DSD-mobile surge com o intuito de oferecer os conteúdos da plataforma *DSD Web* de forma a serem consultados rapidamente através de terminais móveis. Nas situações em que o aluno tem uma posição mais activa, o telemóvel pode ser muitas vezes um terminal de acesso secundário, uma vez que estando fora do contexto escolar (por exemplo em férias) o aluno pode não ter acesso a um computador. A possibilidade de realizar as suas tarefas através de um telemóvel pode ser uma forma de agilizar o processo. No entanto, esta situação verifica-se apenas se o utilizador (aluno) não tiver acesso a um computador, pois isto implica tarefas que limitadas a um ecrã de pequenas dimensões podem ser complicadas de realizar.

No entanto, no segundo momento referido no capítulo anterior (durante o ano lectivo), o aluno torna-se um mero consumidor de informação, portanto, facilitar o processo de acesso a esta informação “fixa” existente na plataforma DSD é uma mais-valia para o aluno. Uma forma de manter os alunos do DETI informados, é dar a possibilidade de, sem ter acesso a um computador, e em qualquer altura, terem a

informação de que precisam, isto é, dar “mobilidade” à informação. O consumo rápido, eficaz e útil no momento justifica a adaptação dessa informação “fixa”.

Neste contexto, o projecto DSD-mobile vem oferecer aos alunos a possibilidade de acederem à informação que precisam em qualquer lugar, tornando-se numa forma de acesso à informação, para muitos, em certas circunstâncias mais vantajosa.

3.2.2 Aquisição e análise de requisitos

Este capítulo apresenta os resultados e discussão dos dados obtidos na componente de captação das necessidades dos utilizadores do sistema DSD e também, da caracterização desses mesmos utilizadores. O instrumento utilizado para a recolha de dados foi o método de inquérito por questionário. Foi realizado um questionário online e enviado por correio electrónico para todos os alunos do DETI (aproximadamente 1000 alunos). O rácio de respostas obtido foi na ordem dos 20% (202 respostas). Esta taxa de respostas representa um valor que se situa dentro do intervalo habitual da taxa de respostas a este processo de inquirição, [10% a 30%] (Tryfos 1996; Sekaran 2003). Questionário semelhante tinha sido anteriormente colocado a 18 docentes do DETI, dos quais se obteve 100% de respostas (o que corresponde também a aproximadamente 20% dos docentes do Departamento). O questionário foi colocado aos potenciais utilizadores (alunos e docentes do DETI) com o objectivo de saber se este sistema seria útil ou não para os mesmos. No entanto, nos pontos seguintes, apenas serão analisadas as respostas dadas pelos alunos, uma vez que os docentes acharam que, para eles, esta plataforma não teria utilidade, pois preferem realizar as suas tarefas num computador. De notar que dos 18 docentes inquiridos, apenas 17% acharam que o sistema móvel poderia ser útil. As perguntas que constituíam o questionário encontram-se no Anexo A.

- **Caracterização dos inquiridos (grupo 1 e 2 do questionário)**

Com o primeiro grupo de questões, pretende-se saber a sua idade, género, estatuto na Universidade de Aveiro e a função que desempenham no caso de serem funcionários. Este grupo de questões permite, *à priori*, saber em termos muito gerais, que tipo de utilização os inquiridos dão à plataforma *web* DSD, de acordo com o estatuto que seleccionarem.

O grupo 2 (Informações Educacionais), trata das informações relativas ao nível académico. Desta forma existe a possibilidade de saber o nível de literacia enquanto competência dependente do percurso escolar.

A amostra deste estudo é constituída por 202 alunos do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro, com idade média de 23 anos, sendo 92% do sexo masculino e 8% do sexo feminino. Este desequilíbrio quanto ao género deve-se ao facto de os cursos existentes no DETI são maioritariamente frequentados por alunos no sexo masculino.

Todas as questões eram de resposta obrigatória à excepção da última que correspondia a sugestões, ou seja, para avançar no questionário os utilizadores tinham que responder a todas as questões de forma sequencial, não podendo avançar no questionário se existissem questões sem resposta.

- **Caracterização quanto à utilização de dispositivos móveis (grupo 3 do questionário)**

O grupo 3, é destinado a conhecer que tipo de utilização os inquiridos dão aos telemóveis. O que se pretende é conhecer a regularidade com que utilizam o telemóvel e quais os serviços que mais costumam utilizar. Este grupo de questões é fundamental, uma vez que a parte prática do projecto de dissertação é assente em dispositivos móveis. Um forte utilizador será, em princípio, um forte utilizador da aplicação DSD-mobile.

Foram colocadas três perguntas, através das respostas à primeira pergunta deste grupo (que tipo de telemóvel possui?), verificou-se que 76% dos inquiridos utilizam um telemóvel que permite acesso à Internet (telemóvel simples, PDA ou *smartphone*), como ilustrado na Figura 23.

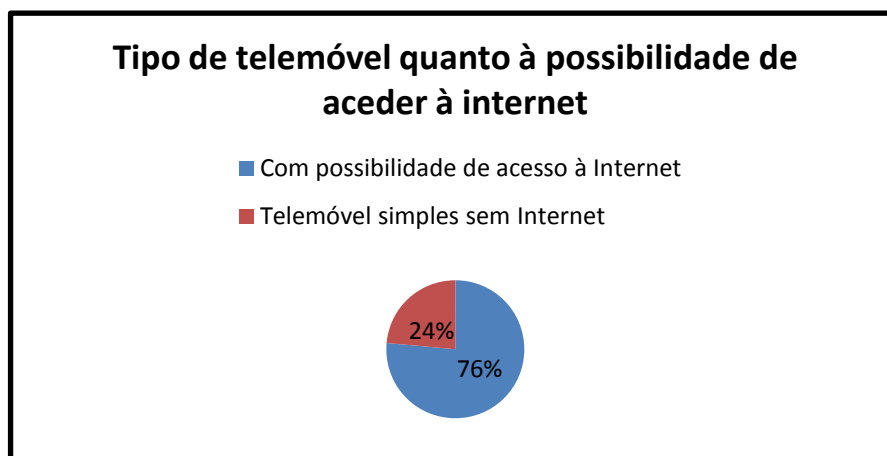


Figura 23 - Resultado do inquérito correspondente ao tipo de telemóvel que o inquirido possui

Dos 76% que têm telemóvel que possibilita acesso à Internet, 66% costumam consultar serviços e páginas *Web* através do telemóvel, como ilustrado na Figura 24.

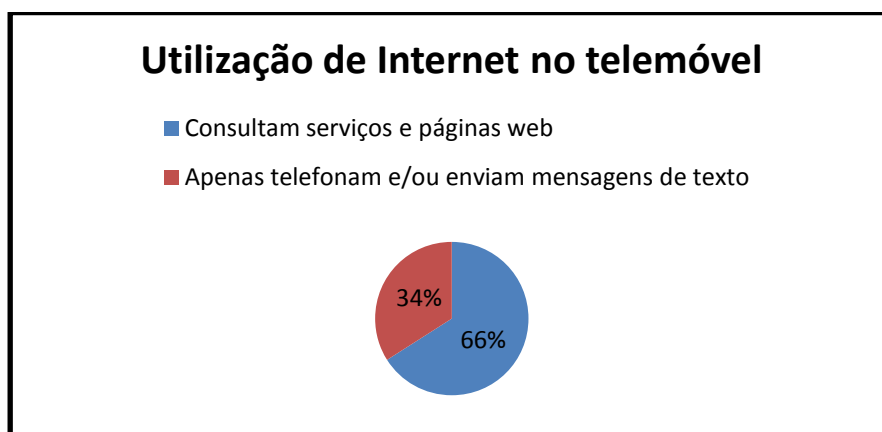


Figura 24 - Resultado do inquérito correspondente à utilização de Internet no telemóvel

Estes resultados indicam que os alunos inquiridos possuem telemóveis com avançada tecnologia e não apenas telemóveis simples sem acesso à Internet (24%) e que costumam aceder à Internet através dos seus dispositivos móveis (66%). Daqui se retira a ideia de que estes poderão ser os potenciais utilizadores da plataforma DSD-mobile.

- **Utilização da plataforma DSD Web (grupo 4)**

O quarto grupo de questões referia-se à utilização da plataforma DSD Web. Uma vez que os potenciais utilizadores da aplicação DSD-mobile são, na sua maioria, utilizadores da plataforma existente, achou-se pertinente incluir questões relativas às funcionalidades existentes, bem como à frequência e tipo de utilização.

Os inquiridos foram abordados com duas perguntas: a frequência com a qual utilizavam a plataforma e para que fins a utilizavam.

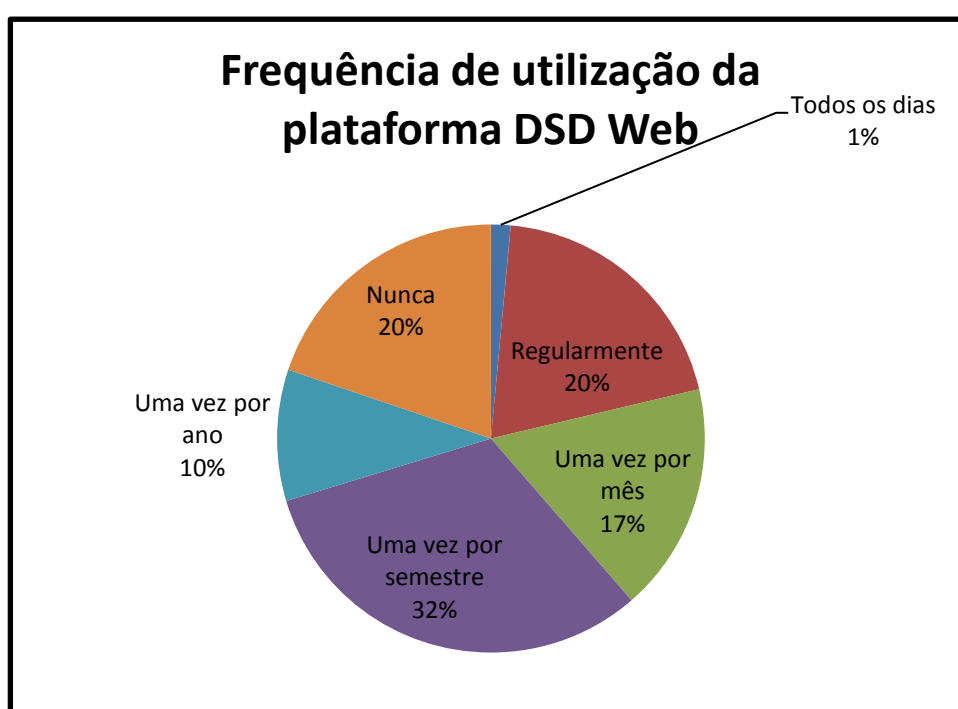


Figura 25 - Resultado do inquérito quanto à frequência de utilização da plataforma DSD Web

Como se pode ver na Figura 25, apenas 1% dos inquiridos visitam a página Web todos os dias. 32% dos inquiridos utilizam a plataforma uma vez por semestre. Este valor deve-se ao facto de no início de cada semestre ser necessário fazer inscrições em disciplinas através da plataforma. Os inquiridos que visitam a página regularmente (20%) utilizam-na, em grande parte, para visualização de conteúdos, *download* de documentos e para fazer inscrições.

- **DSD-mobile (grupo 5)**

Por fim, e afinilando mais o propósito das questões, o questionário inclui um grupo relacionado com a futura aplicação DSD-mobile. Com este grupo de questões, pretendia-se estimar o grau de utilização que possivelmente seria dado à aplicação *Web*. Pretende-se calcular a pertinência do novo sistema face às funcionalidades existentes na plataforma *Web* e às que o inquirido acrescentaria.

Pediu-se ao inquirido que imaginasse uma aplicação *Web* para dispositivos móveis, na qual seriam integradas as funcionalidades mais relevantes da plataforma DSD *Web*. De seguida solicitou-se que respondessem a 3 questões, duas de resposta fechada e uma de resposta livre. A primeira, e a mais importante para este estudo, referia-se à utilidade de acompanhar as actualizações da plataforma DSD *Web* através do telemóvel. A esta pergunta 42% dos inquiridos respondeu positivamente, 36% respondeu que talvez fosse útil, apenas 12% acharam que não seria útil e 11% foi indiferente a esta questão. A Figura 26 ilustra estes resultados. A razão de 36% dos inquiridos estar indeciso com a utilidade é talvez a inexistência de uma plataforma móvel na Universidade como termo de comparação.

Com isto achou-se pertinente dar continuidade ao projecto de adaptação dos conteúdos da plataforma e criar um sistema através do qual o aluno pudesse consultar a informação que necessita numa plataforma móvel.

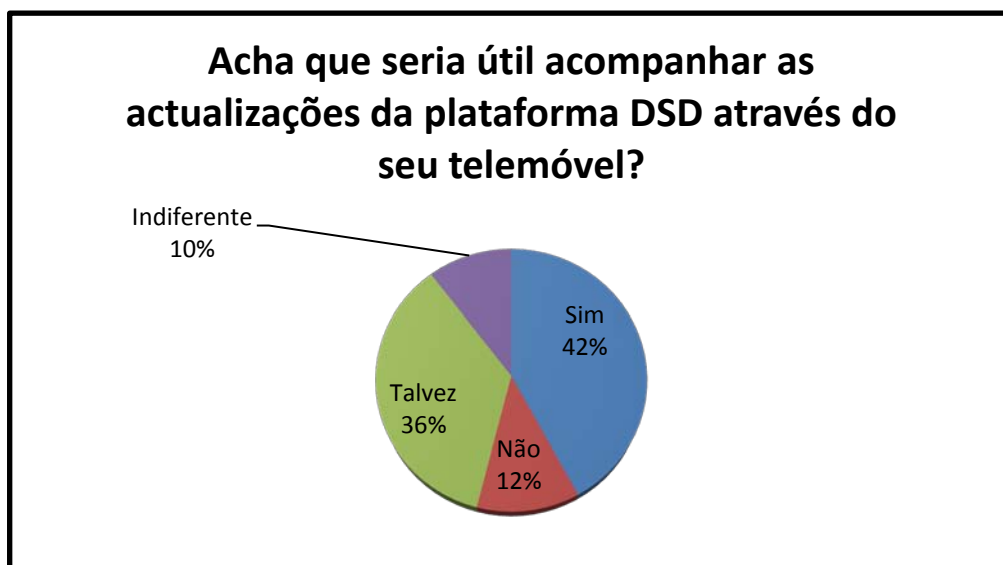


Figura 26 - Resultado do inquérito quanto à utilidade da aplicação DSD-mobile

Uma grande parte dos inquiridos (78%) responderam que achariam pertinente adaptar praticamente todas as funcionalidades existentes no DSD *Web* para dispositivos móveis. Foram adicionadas mais duas opções de resposta correspondentes a duas funcionalidades que ainda não estão em prática no sistema actual: a possibilidade de visualizar, em tempo real, que salas estão livres/ocupadas e quais os docentes que estão disponíveis/ocupados. 71% dos inquiridos, para além das funcionalidades existentes, adicionaria estas duas funcionalidades ao sistema.

A última questão, de resposta livre, não sendo portanto obrigatória de responder, permitia aos inquiridos dar sugestões de funcionalidades que achassem pertinente adicionar à plataforma adaptada a dispositivos móveis. No total foram dadas 17 respostas válidas (cerca de 8,4%).

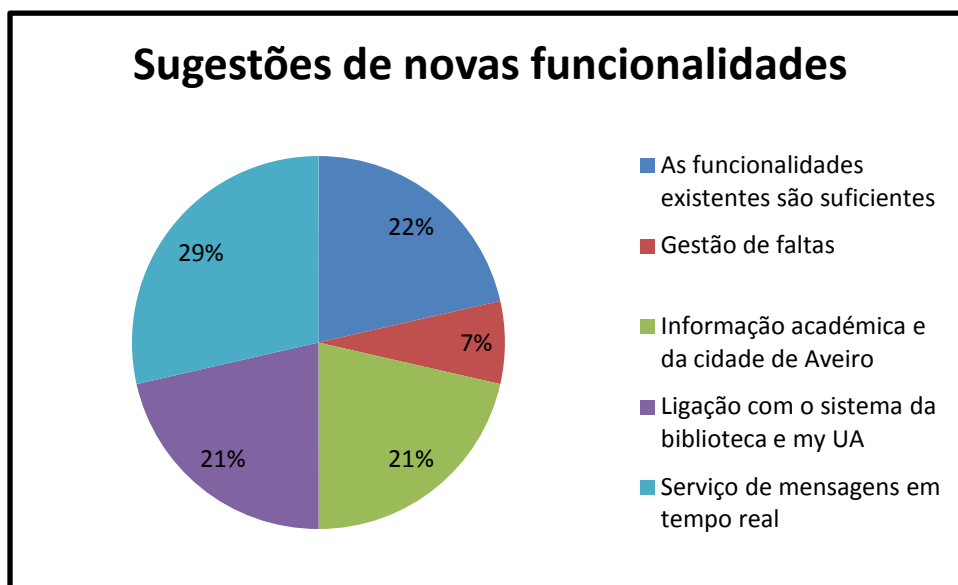


Figura 27 - Sugestões fornecidas pelos inquiridos para a aplicação *mobile*

Como se pode ver na Figura 27, 29% dos inquiridos acharam que seria pertinente adicionar a funcionalidade que permita contactar os docentes e os colegas em tempo real através de um serviço de mensagens instantâneas. No entanto, a reacção dos docentes quando deparados com esta sugestão foi negativa. Os docentes preferem utilizar o serviço de e-mail da Universidade para comunicar com os alunos. 21% achou que seria interessante uma ligação com os serviços da biblioteca para pesquisa de livros através do telemóvel. 21% achou que também seria pertinente adicionar informações acerca da Universidade e da cidade de Aveiro. Um inquirido achou que se poderia integrar um sistema de gestão de faltas para que o aluno saiba, no momento, as faltas que tem a uma disciplina. E finalmente, 22% achou também que as funcionalidades que existem na plataforma *Web* são suficientes e talvez demasiadas para adaptar ao sistema móvel.

3.2.2.1 Conclusão

Como se referiu no início deste capítulo, as principais motivações desta componente do estudo (através da recolha e análise de dados pelo inquérito baseado em questionário) agrupam-se de certa forma em três categorias: (1) obter um conjunto de dados que permita caracterizar quanto ao tipo de utilizadores da plataforma DSD *Web* e

quanto à utilização de telemóveis com acesso à Internet; (2) perceber a posição dos inquiridos perante a plataforma *DSD Web* existente; (3) aferir o interesse e motivação, através da utilidade que os inquiridos atribuem a uma eventual adaptação das funcionalidades do sistema para dispositivos móveis, já que representam um grupo muito importante de utilizadores.

Quanto a (1) verificou-se que os utilizadores são alunos do DETI, com idade média de 23 anos e, na sua maioria, são do sexo masculino. Cerca de 66% são portadores de telemóveis com acesso à Internet e costumam consultar serviços ou páginas *Web* através do telemóvel. Confirmando-se assim que são potenciais utilizadores do novo sistema e reúnem as condições básicas para utilização de uma aplicação *Web* adaptada a dispositivos móveis.

Quanto a (2) na sua maioria utilizam a plataforma *Web* uma vez por semestre para inscrições e visualização de conteúdos. Dizem conhecer bem a plataforma *Web* e dar utilização a praticamente todas as funcionalidades de que dispõe.

No que diz respeito a (3), confirma-se o interesse em utilizar a plataforma adaptada a dispositivos móveis. A aceitação do sistema parece estar confirmada. De uma forma generalizada atribuíram grande utilidade ao sistema *DSD-mobile* (41% dos que possuem telemóvel com Internet), defendendo a adaptação das funcionalidades da plataforma *Web* para dispositivos móveis. Foram adicionadas duas funcionalidades novas, que foram seleccionadas para adicionar ao novo sistema por 71% dos utilizadores: visualização em tempo real das salas que estão livres/ocupadas e docentes que estão a dar aulas ou não.

Tendo em conta os resultados obtidos, considera-se que existem fortes indícios da viabilidade de implementação da solução em questão.

3.2.3 Aplicação Web versus Nativa

No início do desenvolvimento da plataforma *DSD* adaptada a dispositivos móveis, surgiu uma questão fundamental: desenvolver uma aplicação *Web* ou uma aplicação nativa?

Teve então que se analisar as necessidades reais do sistema. Estudaram-se os pontos positivos e negativos de cada tipo de aplicação para daí chegar a uma resposta.

Como foi referido no capítulo 2, as aplicações nativas são necessariamente desenvolvidas especificamente para as plataformas dos dispositivos em questão. Uma desvantagem deste tipo de aplicações seria a necessidade de instalação no dispositivo. Sendo que a maioria das aplicações são certificadas, vendidas e distribuídas através de uma loja online, seria necessário (pensando no caso de utilizadores do iPhone) colocar a aplicação na loja online da Apple. Noutros casos em que a certificação de aplicações não é tão rígida, e dado que estas aplicações assentam sobre a plataforma do dispositivo podem aceder a funcionalidades do mesmo podendo originar falhas de segurança. A vantagem deste tipo de aplicações é que se pode tornar mais fácil de implementar para uma única plataforma. No entanto, os dispositivos não possuem todos o mesmo sistema operativo, tendo que existir uma forma da plataforma se adaptar aos mais variados dispositivos móveis.

Pensou-se então na implementação de uma aplicação *Web*, considerando as suas vantagens e desvantagens relativamente às aplicações nativas. O facto de não necessitar instalação e de poder ser acedida através de qualquer *browser*, foram as razões determinantes para se optar por uma aplicação *Web*. O ponto mais negativo que apresenta, é o facto de consumir tráfego de Internet, no entanto quando analisados todos os pontos positivos, verifica-se que uma aplicação *Web* será a melhor opção de implementação.

3.2.4 Design

- *Layout*

O *layout* da aplicação foi construído tendo em conta o fluxo de informação desejado numa página *Web* desenhada para dispositivos móveis referido por Mehta (2008). No início da página temos o *header* ou cabeçalho, a seguir o painel de navegação, segue-se o conteúdo, novamente o painel de navegação e por último o rodapé (como mostrado na Figura 28).

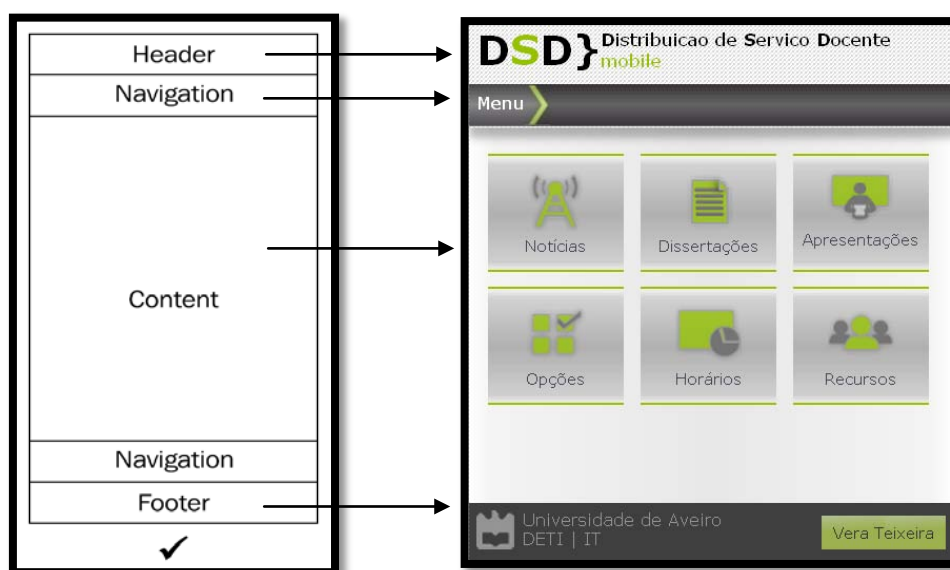


Figura 28 - Comparação entre o fluxo de informação desejável e o fluxo existente na aplicação DSD-mobile

Como se pode verificar, o DSD-mobile segue a estruturação de informação desejável à excepção do segundo painel de navegação. Considerou-se que um segundo painel no final da página seria redundante, uma vez que os passos para passar de um menu para outro são mínimos e também, porque a quantidade de informação por página é relativamente reduzida, o que possibilita na maior parte das vezes, ter o painel superior visível.

- **Tamanho de ecrã**

A aplicação DSD-mobile foi otimizada para dispositivos com tamanho de ecrã 480x320 *pixels* (3,5 polegadas). Foi tomada esta opção porque, como foi visto inicialmente, os ecrãs dos telemóveis têm vindo a aumentar cada vez mais. Telemóveis como o iPhone, Samsung Galaxy S, Google Nexus One, HTC Desire, Motorola Milestone são relativamente recentes no mercado e o seu tamanho de ecrã em polegadas ronda as 3,5. Esta tendência justifica a optimização da aplicação para este tamanho de ecrã.

Quanto à resolução definida para a aplicação, considerou-se mais adequado respeitar a largura anteriormente referida (320 *pixels*). No caso da altura optou-se por esta ser variável tendo, no entanto, um tamanho mínimo de 480 *pixels*.

Devido a esta característica torna-se necessário repensar as várias opções de design para inclusão de cada um dos conteúdos da solução. Possivelmente, será necessário adaptar-se a estrutura de algumas áreas para que o conteúdo nestas incluído se ajuste ao tamanho de ecrã de cada um dos dispositivos.

- **Zoom**

Muitos *browsers* de *smartphones* utilizam a largura mais adequada para a exibição das páginas *Web* permitindo que o utilizador faça *zoom in* e *zoom out*. A *viewport metatag* permite que o *developer* defina os melhores limites da página *Web* que desenvolve. Esta *tag* controla a largura, altura e o tamanho inicial da página *Web*. Tem, ainda, a função de especificar o *zoom* máximo e mínimo que o utilizador pode fazer à aplicação. A sua presença indica também que a página é optimizada para dispositivos móveis (Frederick e Lal 2009).

No projecto do DSD-*mobile* foi utilizada a *metatag viewport* da seguinte forma:

```
<meta name="viewport" content="width=device-width; user-scalable=yes; initial-scale=1; maximum-scale=2;" />
```

Em que a directiva *width* especifica que a largura da aplicação ajustará à largura do dispositivo (*width=device-width*). A directiva *user-scalable* define se o utilizador tem permissão ou não para fazer *zoom in* ou *zoom out* à janela. Neste caso, foi dada esta possibilidade (valor “yes”) uma vez que em dispositivos com ecrãs maiores, por exemplo no iPad, a aplicação ficaria bastante pequena e por vezes o texto tornava-se imperceptível. Para além de que se pode aproveitar melhor a largura do ecrã, quando o iPad se encontra na posição horizontal. A directiva *initial-scale* define o factor de multiplicação de *zoom*, que neste caso, estando com o valor 1 significa que se vai adaptar a 100% ao tamanho do objecto anteriormente definido (480x320 *pixels*). A directiva *maximum* define o limite máximo para o dimensionamento da aplicação. Com o valor a dois significa que o limite do *zoom* máximo é duas vezes o tamanho definido para o objecto. Em vários testes com diferentes dispositivos de diferentes tamanhos de ecrã verificou-se que quando se faz *zoom in* até ao limite se consegue visualizar todos os conteúdos de uma forma bastante aceitável.

- **Iconografia**

Como se pode ver na Figura 29, ao navegar na plataforma DSD verifica-se que esta contém muito poucos ou quase nenhuns ícones. Talvez esta característica não seja tão importante assim quando o *site* é exibido num computador, pois os menus são totalmente descritivos contendo apenas texto que permite perceber à partida o que se pretende em cada área. No entanto, quando se desenha uma interface para dispositivos móveis convém acompanhar a descrição de um menu com um ícone. Isto torna-se importante porque devido ao tamanho reduzido do ecrã, o texto pode não ser totalmente visível ou o utilizador ter dificuldades em perceber o que está escrito. Um ícone pode ser uma forma de o utilizador perceber rapidamente e sem muitas dificuldades o que se pretende, ligando uma representação gráfica a um conceito.

No desenho dos ícones utilizados na aplicação procurou-se compreender os conceitos por trás dos processos e chegou-se a possíveis representações apresentadas na Figura 30.

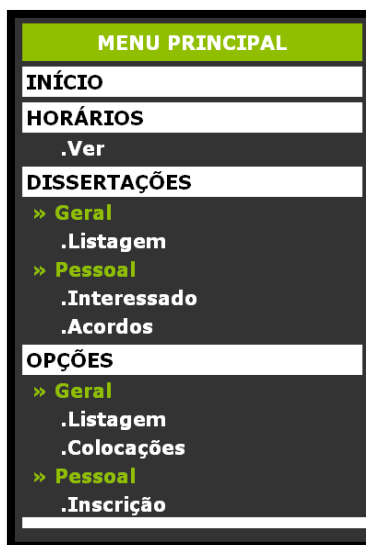


Figura 29 - Ausência de ícones no menu da plataforma Web

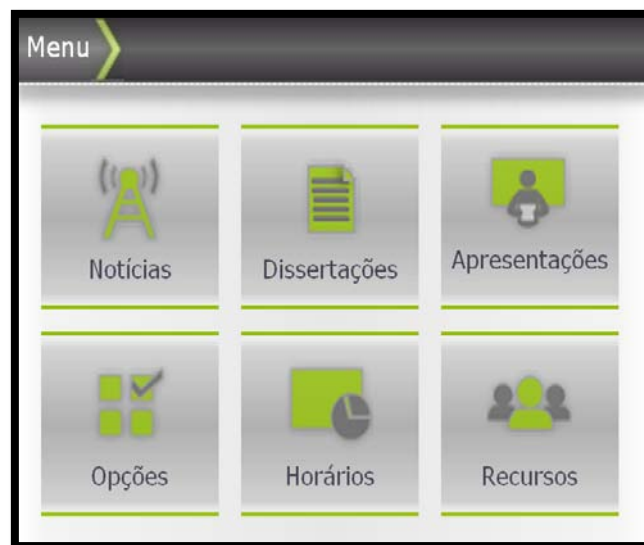


Figura 30 - Ícones utilizados no menu da aplicação DSD-mobile

- **Áreas de interacção/botões**

Ao otimizar uma plataforma para dispositivos móveis, é muito importante ter em conta as dimensões reduzidas dos ecrãs e consequentemente a maior dificuldade em visualizar o conteúdo. As áreas de interacção devem ser perceptíveis de modo a que o utilizador saiba à partida que se trata de uma área associada a uma determinada funcionalidade.

No que diz respeito aos botões, para se obter uma melhor experiência do utilizador, optou-se por desenhar botões grandes e simples (*design* minimalista). Existem quatro tipos de botões na aplicação: botão de lista, de menus, de navegação entre áreas e botões de saída (como se pode ver na Figura 31).

Imagem	Descrição
	<p>Botões do menu principal. Identifica uma área importante do sistema.</p>

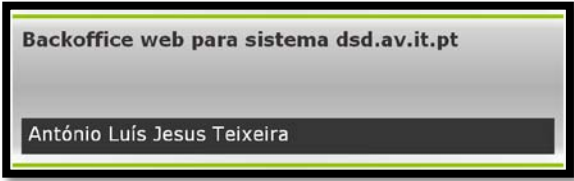


	<p>Botão de lista. Existente nas áreas que contêm listagens. Ao ser seleccionado aparece informação detalhada acerca do assunto em questão.</p>
	<p>Botões rápidos de navegação entre áreas principais.</p>
	<p>Botões de saída de uma área. Quando é seleccionado volta à área anterior.</p>

Figura 31 - Diferentes tipos de botões existentes na aplicação DSD-mobile

Basicamente a aplicação é constituída por diferentes áreas e a forma para chegar até elas é através dos botões existentes, pelo que devem ser bem visíveis e fáceis de seleccionar. Optou-se por botões de maiores dimensões, facilitando assim a sua utilização através do toque.

- **Imagens**

Quanto às imagens utilizadas, tentou-se que estas ficassem num formato que não pesasse muito no carregamento da página. Imagens pequenas seriam a melhor solução, no entanto, foi necessário ter em conta que tamanhos demasiado reduzidos poderiam prejudicar a visibilidade e compreensão.

Foram utilizadas imagens nas áreas Docentes e Notícias, em ambos os casos as imagens têm um tamanho de 60x60 *pixels*. Tamanho este, que se considerou ser o mais adequado quando comparado com o tamanho da interface da aplicação e a área disponível.

- **Ajudas à navegação (*breadcrumbs*)**

Um *breadcrumb* é um tipo de esquema de navegação auxiliar que revela a localização do utilizador num Website ou numa aplicação. No caso da aplicação

DSD-mobile foi utilizada uma área de *breadcrumbs* no painel de navegação, como mostrado na Figura 32.

A utilização deste esquema de navegação auxiliar oferece aos utilizadores uma forma de, dentro de uma hierarquia, ter uma visão geral e estruturada do ponto em que se encontra. A aplicação *Web* é uma aplicação que se encontra organizada de uma forma hierárquica portanto a utilização deste tipo de esquema de navegação é aconselhável.



Figura 32 - Exemplo do esquema de navegação *breadcrumb* na aplicação

- **Cores**

As cores utilizadas estão de acordo com a página *Web* existente. Para haver consistência foram utilizados os mesmos padrões: verde, cinza, preto e branco.

A sua qualidade e semelhança com a plataforma depende em muito do dispositivo utilizado. Cada dispositivo pode mostrar uma cor de forma diferente o que faz com que a mesma cor possa parecer diferente em dispositivos diversos. Os códigos das cores utilizados foram os mesmos que na plataforma *DSD Web*, portanto as diferenças ao nível dos tons das cores detectadas entre ambas (página *Web* e aplicação móvel), dependerão do dispositivo e das condições de luminosidade a que estiver exposto.

- **Listas**

Em determinadas áreas foi necessário repensar a estruturação de conteúdos, nomeadamente na área de Notícias, Dissertações, Apresentações e Opções. Estas áreas continham bastante informação tornando-se a sua adaptação complicada.

Optou-se por utilizar uma lista de botões, em que a selecção de cada botão apresenta os detalhes do que se pretende. Esta separação da informação principal dos detalhes foi feita de forma a minimizar a informação contida em cada página. Tornar-se-

ia demasiado maçador para o utilizador ter a informação de cada dissertação numa mesma página, se este estivesse à procura de uma dissertação em concreto.

3.2.5 Tecnologias de implementação

As tecnologias de suporte ao desenvolvimento da aplicação DSD-*mobile* foram essencialmente ASP.NET e a linguagem de suporte foi C#.

O desenvolvimento *Web* para dispositivos móveis em ASP.NET não difere em muito do desenvolvimento *Web* para *desktop*. O ASP.NET fornece o *namespace* <System.Web.Mobile> que é dedicado ao desenvolvimento *Web mobile*. No entanto, este conjunto de classes não foi utilizado na implementação do projecto DSD-*mobile* uma vez que, logo de início se definiu quais os *browsers* para que se iria desenvolver a aplicação (*browsers* do tipo Webkit). Isto porque as principais plataformas existentes no mercado hoje em dia (iOS, Android, Symbian e BlackBerry) possuem este motor de renderização. À partida, este tipo de dispositivos móveis suportariam aplicações neste formato, obviamente com os ajustes referentes às limitações dos telemóveis quando comparados com um *desktop*. Embora o processo de desenvolvimento de páginas *Web* seja o mesmo para *desktop* e dispositivos móveis, a arquitectura ASP.NET obrigada a uma implementação cuidada. Para uma aplicação ser exibida em telemóveis terá de sofrer os devidos ajustes de forma a maximizar a compatibilidade com dispositivos que possuam diferentes capacidades de processamento, tamanhos de ecrã e limitações de *browsers*, bem como diferentes resoluções.

Existem algumas razões para o facto de se ter utilizado ASP.NET em detrimento, por exemplo, de PHP. As razões foram essencialmente as seguintes:

- **Aproveitamento de código** - A plataforma DSD *Web* foi implementada utilizando esta tecnologia, o que possibilitou um aproveitamento do código existente, não só nas consultas à base de dados mas também no aproveitamento de funções já implementadas. Daí também se ter utilizado a linguagem C#.
- **Integração de tecnologias** – A utilização de ASP.NET permitiu a integração directa com as tecnologias utilizadas no servidor de alojamento. O facto da base de dados DSD *Web* estar armazenada no

SGBD SQLServer facilita a integração com ASP.NET pois ambas as tecnologias são produto da Microsoft, tendo portanto compatibilidade directa.

- **Possibilidade de implementação da lógica MVC (Model View Controller)** – Este tipo de arquitectura de desenvolvimento de aplicações possibilita que uma mesma solução se ajuste a diferentes suportes de execução. No contexto de aplicações *Web*, um só endereço *Web* pode permitir aceder a diferentes versões de uma mesma aplicação mas em diferentes formatos, ajustando-se à plataforma (computador ou dispositivo móvel) onde é executada. Na génese desta arquitectura está a possibilidade de partilha de todas as rotinas de obtenção de dados, sendo estas invocadas por cada uma das versões da aplicação. Ao ser invocada uma aplicação construída segundo este modelo, o servidor detecta qual é a plataforma na qual a aplicação será executada, faz o processamento de dados e envia ao cliente a versão adequada. Com apenas um projecto implementado pode-se conseguir diferentes formatações da aplicação. Basicamente o código C# (*c-sharp*) é o mesmo, alterando o código referente à formatação da interface. No entanto, no projecto da aplicação *DSD-mobile* essa lógica é demonstrada de outra forma. Foi criado um endereço de teste para demonstrar esta funcionalidade. Quando este endereço é colocado num *browser mobile* o servidor redirecciona para o endereço da aplicação *DSD-mobile*, por outro lado, quando o endereço é colocado num *browser PC* a página devolvida é a *DSD-Web*.

3.2.6 Testes à aplicação

A aplicação foi testada em quatro dispositivos reais (indicados na Tabela 8), em simuladores PC e através de ferramentas de teste online.

Foram escolhidos estes dispositivos de teste porque possuem cada um dos sistemas operativos mais utilizados actualmente (Gartner 2010). Relativamente aos testes em simuladores e ferramentas online, optou-se pelas indicadas porque representam o tipo de *browsers* em que aplicações deste tipo podem ser consultadas.

- **Dispositivos físicos**

Dispositivo	Sistema Operativo	<i>Browser</i>	Mecanismo de renderização
iPhone	iOS	Safari	Webkit
TMN a1	Android	<i>Android browser</i>	Webkit
Nokia E52	Symbian 9.3	Nokia Minimap	Webkit
TMN bluebelt	Windows <i>Mobile</i> 6.1	Internet Explorer 6	Proprietário (<i>Trident</i>)

Tabela 7 - Dispositivos físicos utilizados para teste da aplicação

Nos *browsers* cujo motor de renderização é Webkit a aplicação comportou-se de forma semelhante. Uma vez que o mecanismo de renderização é o mesmo, os *browsers* executaram a aplicação da mesma forma. O motor Webkit fornece um conjunto de classes para exibir o conteúdo de uma página *Web* e implementa funcionalidades no *browser* que fazem com que as páginas sejam executadas da mesma forma, reconhecendo o *markup* de forma semelhante. Por estes motivos, a aplicação *Web* não sofreu alterações nem ficou desformatada nos dispositivos com este tipo de motor. Nestes testes existiu um problema evidente: no telemóvel Nokia E52 a interface da aplicação apareceu com *scroll* horizontal e vertical. Este problema deve-se ao facto deste dispositivo possuir um ecrã ligeiramente mais pequeno que o tamanho definido no código (480x320 *pixels*) e à directiva *initial-scale* (com o valor 1) da *metatag viewport*, que indica que, independentemente do tamanho do ecrã do dispositivo, a aplicação irá ser exibida em 100% do tamanho mínimo definido (480x320 *pixels*). As diferenças mais significativas na execução da aplicação foram a nível das cores e da velocidade de processamento da página. Estas características são fortemente condicionadas pelo tipo de ecrã e capacidade de processamento dos dispositivos.

Já no caso do dispositivo TMN bluebelt, a experiência não foi a mesma. O *browser* não interpretou o código da mesma forma devido ao mecanismo de renderização ser diferente. Aconteceram alguns problemas de formatação dos conteúdos

e alterações dos tamanhos dos objectos. Conclui-se então que o Internet Explorer Mobile apresenta algumas fragilidades de interpretação de alguns dos estilos CSS, bem como na execução de partes do javascript gerado automaticamente pela compilação .NET.

- **Simuladores**

Encontram-se na Tabela 9 os simuladores utilizados durante o desenvolvimento da aplicação DSD-mobile.

Simulador	Mecanismo de renderização
Opera Mini	Opera Presto
Firefox for <i>mobile</i>	Gecko
BlackBerry Torch 9800	Webkit
Skyfire	Gecko
iPad Peek	iPhone OS 3

Tabela 8 - Simuladores utilizados nos testes da aplicação

A primeira experiência foi realizada no Opera *Mobile*, um simulador que pode ser descarregado, estando disponível para Windows, Linux e MacOS. Simula as aplicações *Web* como se se estivesse a utilizar um telemóvel. Quando o *browser* carrega a aplicação, esta aparece com um tamanho bastante reduzido. Testou-se a aplicação no simulador e em vários dispositivos, mas o problema manteve-se. Considerou-se que o problema pudesse ser gerado pela interpretação incorrecta da directiva *width* na *metatag viewport* (*width=device-width*), então experimentou-se atribuir à directiva *width* a largura do ecrã em polegadas, ou seja *width=320*. Fizeram-se vários testes no *browser* Opera Mini e, de facto, esta alteração fez com que a aplicação fosse exibida no seu tamanho normal (320x480 *pixels*).

Em todos os testes com os simuladores Fennec, BlackBerry, Skyfire e iPad Peek, a aplicação comportou-se como esperado. O código Javascript e CSS foram interpretados como se pretendia, sem desformatações ou erros de execução.

- **Ferramentas online**

MobiReady⁴⁹ é uma ferramenta online de avaliação de *Websites* construídos para serem exibidos em dispositivos móveis, tendo em conta padrões e as boas práticas de desenvolvimento. Esta ferramenta permite que o programador introduza o endereço da sua aplicação *Web*, devolvendo um relatório detalhado acerca de algumas especificidades do *Website* e atribuindo uma nota de 0 a 5, em que 0 corresponde a mau e 5 a muito bom. São avaliadas características como o tamanho total da página, custo de desenvolvimento e tempo de processamento da página *Web*. A nota atribuída à aplicação foi de 3. Este valor deve-se ao facto de algumas páginas como as áreas “Dissertações”, “Notícias” e “Horários” conterem listagens com bastante informação. Foi por este motivo que se optou por colocar apenas as listas correspondentes ao ano lectivo corrente, ao contrário do DSD *Web* que conta com as listagens de três anos lectivos na área de “Dissertações”.

O mecanismo **W3C mobileOK Checker**⁵⁰ executa vários testes numa página *Web* para determinar o seu nível de facilidade de utilização em dispositivos móveis. Uma página *Web* está “*mobileOK*”⁵¹ quando passa todos os testes a que se propõe.

Nos testes à aplicação desenvolvida, a ferramenta online gerou uma percentagem de avaliação e um relatório detalhado com todos os erros e uma atribuição do grau de gravidade de cada um (Muito grave, grave, médio e pequeno). O resultado obtido na avaliação principal foi de 57% *mobileOK*. Esta nota geral deve-se principalmente ao tipo de erros encontrados na verificação: um erro grave e seis erros pequenos.

Relativamente ao erro apontado como grave, o relatório indica que se trata de código javascript justificando que nem todos os *browsers* interpretam este tipo de

⁴⁹ <http://mobiready.com/>

⁵⁰ <http://validator.w3.org/mobile/>

⁵¹ *mobileOK* é uma expressão utilizada pela W3C (2010) que significa que um website reúne as condições necessárias para ser executado da melhor forma em dispositivos móveis.

código, podendo gerar uma mensagem de erro ao utilizador, provocando impacto na experiência do utilizador. No entanto nos testes que se fizeram anteriormente nos dispositivos e nos simuladores, a aplicação foi totalmente aceite, ou seja, os scripts foram bem interpretados pelos *browsers*.

Os erros considerados de baixa relevância, estão relacionados com:

- Tamanho total da aplicação: A aplicação ocupa 38,8KBytes, ultrapassando em 20KBytes o tamanho ideal. Isto deve-se à quantidade de imagens existentes nas áreas de “Notícias”, “Docentes” e “Salas”. No entanto, nos diversos testes o tamanho da aplicação não comprometeu demasiado o carregamento da página;
- Código CSS: devem-se principalmente à estruturação do código. Este problema apenas se evidencia nos ficheiros de código, não se evidenciando quando a aplicações é executada.

4 Avaliação de Usabilidade da aplicação

Um dos factores mais importantes de um sistema é a forma como os utilizadores irão comunicar com ele. A interface deve ser de fácil aprendizagem e deve ser necessário realizar poucos passos para atingir um objectivo.

Para se avaliar a usabilidade de um sistema, usam-se métodos de engenharia de usabilidade que permitem elaborar testes. Esses testes, segundo Nielsen (1993) consideram se o sistema é fácil de utilizar, de aprender, de memorizar, se não têm erros e se causam satisfação ao utilizador.

Apesar de em qualquer sistema, os testes de usabilidade serem indispensáveis, no contexto móvel são particularmente importantes devido às limitações dos dispositivos. Obviamente que o utilizador espera ter uma experiência semelhante ou até melhor quando utiliza outro tipo de dispositivo.

Normalmente considera-se que um sistema tem problemas de usabilidade quando os utilizadores encontram dificuldades na realização de tarefas e quando demoram mais tempo que o desejado a concretizá-las.

Para garantir uma melhor utilização da aplicação desenvolvida foram realizados testes de usabilidade de forma a detectar possíveis problemas existentes. Através destes testes foi possível recolher algumas sugestões para melhorar a aplicação e assim torná-la mais funcional e atractiva.

4.1 Métodos utilizados

A avaliação de usabilidade é uma necessidade para a viabilidade de um sistema, pois os utilizadores vão querer evitá-lo caso a sua utilização seja complexa. A aplicação *DSD-Mobile* não é excepção. Foram realizados testes tendo como base dois métodos: a avaliação heurística e os testes de usabilidade com questionário final.

4.1.1 Avaliação Heurística

A avaliação heurística permitiu identificar falhas a nível da interface de utilizador. Permitiu de uma forma rápida e com custos reduzidos identificar problemas de design.

Optou-se por utilizar as dez heurísticas de Jakob Nielsen (2000b) uma vez que estas são adequadas para a análise de uma interface *Web*, e portanto são adequadas também para avaliar a aplicação DSD-mobile.

A avaliação heurística, sendo um método subjectivo, foi realizada por um conjunto de quatro avaliadores com alguma experiência de utilização deste método. Estes examinaram a interface individualmente de forma a serem detectados mais facilmente os vários problemas existentes na interface. Para cada problema encontrado foi definido um grau de gravidade que define a importância de resolução do mesmo para que possa ser corrigido para um melhor funcionamento do sistema. Considerou-se que é um número aceitável porque, segundo Nielsen (1993), cinco avaliadores tem sido um número recomendável, tendo em consideração que um aumento no número de utilizadores não produz a identificação de muito mais problemas de usabilidade.

Cada avaliador teve a oportunidade de experimentar o sistema durante algum tempo. Quando se sentiu preparado avaliou o sistema segundo as dez heurísticas enunciadas anteriormente. Numa escala de 0 a 4, em que o valor 4 significa um problema muito grave, os avaliadores quantificaram, no seu ponto de vista, qual a gravidade de cada problema encontrado.

4.1.1.1 Resultados

Os resultados obtidos foram bastante satisfatórios. Os avaliadores apontaram apenas problemas com grau de gravidade 0 1 e 2.

Como se pode verificar pela Tabela 10, devem fazer-se pequenos ajustes na aplicação, no entanto aos problemas que foram apontados foi atribuído maioritariamente o grau 1, ou seja, são problemas que não afectam o funcionamento do sistema e portanto não necessitam de ser solucionados imediatamente.

Os principais problemas encontrados são relativos às heurísticas:

- Controlo e liberdade do utilizador: na tarefa de fazer *logout* da aplicação, os utilizadores tiveram alguma dificuldade devido a terem que entrar na área de perfil para terminarem a sessão.
- Flexibilidade e eficiência de utilização: os utilizadores acharam que as tarefas são facilitadas para utilizadores mais experientes. É necessário ter algum tipo de experiência no tipo de dispositivos e aplicação.
- Ajuda e documentação: os utilizadores sentiram falta de algumas ajudas durante a navegação entre as áreas e em tarefas mais complexas.

Heurística	Descrição	Grau de gravidade atribuído por cada avaliador				Mediana do grau de gravidade atribuído
Visibilidade do estado do sistema.	Se o sistema informa o utilizador do que está a fazer a qualquer momento.	0	0	1	0	0
Correspondência entre o sistema e o mundo real.	Se a linguagem é adaptada ao utilizador.	0	0	1	0	0
Controlo e liberdade do utilizador.	Se o utilizador tem o poder de controlar o sistema podendo abandonar uma tarefa a qualquer momento.	1	1	0	1	1
Consistência e Standards.	Se os comandos com o mesmo objectivo se situam sempre no mesmo local e se a sua funcionalidade é sempre a mesma.	0	0	1	1	0,5
Prevenção de erros.	Se evita situações de erro.	0	1	0	0	0
Reconhecimento <i>versus</i> lembrança.	Evitar que o utilizador se tenha que lembrar de algo para executar uma acção.	0	0	0	1	0

Flexibilidade e eficiência de utilização.	Se as tarefas são facilitadas aos utilizadores mais experientes, pela utilização de atalhos, evitando perdas de tempo desnecessárias.	1	1	0	1	1
Estética e design minimalista.	Se os diálogos são simples e naturais e se a informação necessária é apresentada no	0	0	0	0	0
Reconhecimento e recuperação de erros.	Ajuda a diagnosticar, recuperar e corrigir erros.	0	1	0	1	0,5
Ajuda e documentação.	Se as ajudas estão presentes quando são necessárias.	2	1	1	0	1

Tabela 9 - Grau de gravidade de cada uma das heurísticas

4.1.2 Teste de usabilidade

Na segunda fase de avaliação, colocou-se em prática a realização de testes de usabilidade.

Esta técnica é utilizada para avaliar um produto testando-o com avaliadores representativos dos utilizadores do sistema. Sendo assim, os testes de usabilidade foram realizados por alunos do Departamento de Electrónica, Telecomunicações e Informática. Para estes testes foram considerados sete utilizadores.

Estes testes tiveram como objectivo não só identificar problemas de usabilidade mas também permitir recolher sugestões de melhoramento da aplicação e das funcionalidades nela incluídas, bem como o nível de satisfação dos utilizadores face ao sistema. Os utilizadores tiveram que realizar um conjunto de tarefas comuns enquanto o observador tirou notas acerca do seu comportamento na realização das tarefas. As tarefas foram definidas tendo em conta a sua importância na plataforma *DSD Web*, ou seja, são aquelas que são utilizadas com maior frequência e cujo grau de importância é

mais elevado perante os objectivos dos utilizadores. A seguir apresenta-se a lista de tarefas utilizada.

1. Visualize a listagem das disciplinas de opção.
2. Faça login na aplicação com as seguintes credenciais:
E-mail: utilizador1@ua.pt | **Palavra-passe:** util1
3. Procure as dissertações com os seguintes títulos: “Privacidade no comércio electrónico” e “Análise técnico-económica de redes e serviços de telecomunicações: ferramentas de cálculo” e mostre o seu interesse nas duas.
4. Visualize as dissertações nas quais se encontra interessado.
5. Inscreva-se nas disciplinas de opção com a seguinte ordem de preferência: 1- Interfaces Humano - Computador, 2-Laboratório de Redes, 3-Comunicações Ópticas.
6. Volte ao Menu. Depois verifique que salas do 2º piso do DETI estão ocupadas neste momento.
7. Visualize o seu perfil.
8. Verifique se o docente António Luís Jesus Teixeira está disponível neste momento.
9. Visualize a página do docente referido anteriormente.
10. Visualize o horário do curso Mestrado Integrado em Engenharia de Computadores e Telemática do 2º ano, 1º semestre.
11. Efectue logout, para sair da aplicação.

Durante o teste foi pedido aos utilizadores que registassem a facilidade de execução de cada tarefa enquanto o observador registava algumas características observadas, tais como: se completou a tarefa e em quanto tempo, se cometeu erros durante a sua execução, se necessitou de ajuda para atingir o objectivo da tarefa e o grau de facilidade observado de acordo com a escala **1-Nada fácil** até **5-Muito fácil**. Ao

longo da avaliação da aplicação foram também observados, e devidamente anotados, os comportamentos e comentários/sugestões do utilizador.

No final do teste foi pedido aos utilizadores que respondessem a um questionário, baseado no QUIS (*Questionnaire for User Interaction Satisfaction*) (Shneiderman e Plaisant 2005).

Os formulários utilizados nos testes de usabilidade encontram-se disponíveis no Anexo B desta dissertação.

4.1.2.1 Resultados

Os resultados deste teste foram também, no geral, bastante satisfatórios. Esta afirmação é suportada, por um lado, pelos comentários bastante positivos, dos utilizadores ao longo do teste, por outro, pela análise dos resultados da Tabela 11.

Tarefa	Completo	Cometeu Erros		Ajuda	Facilidade (mediana)	
		Poucos	Muitos		Utilizador	Observador
1	7	0	0	0	5	5
2	7	0	0	0	5	5
3	7	2	0	0	4	4
4	7	0	0	0	5	5
5	7	0	0	0	5	5
6	7	0	0	1	5	5
7	7	2	0	0	5	4
8	7	0	0	0	5	5
9	7	0	0	0	5	5
10	7	0	0	0	5	5
11	7	0	0	0	5	5

Tabela 10 - Resultados do teste de usabilidade

Analisando os resultados presentes na Tabela 11, verifica-se que todas as tarefas foram concluídas por todos os utilizadores. No entanto, existem algumas tarefas em que os utilizadores cometeram alguns erros. Essas tarefas correspondem à procura das

dissertações e demonstração de interesse nestas (tarefa 3) e à visualização do perfil (tarefa 7). Um utilizador solicitou ajuda na tarefa 6, que corresponde a visualizar as salas que estão ocupadas no segundo piso do DETI.

Na Figura 33 pode ser observado o nível de facilidade atribuído, pelos utilizadores e pelo observador, a cada tarefa. Verifica-se que, tanto os utilizadores como o observador, atribuíram um nível de facilidade bastante alto em praticamente todas as tarefas.

Com isto, pode então concluir-se que a interface da aplicação possui um nível de usabilidade aceitável, não sendo portanto necessários ajustes urgentes.

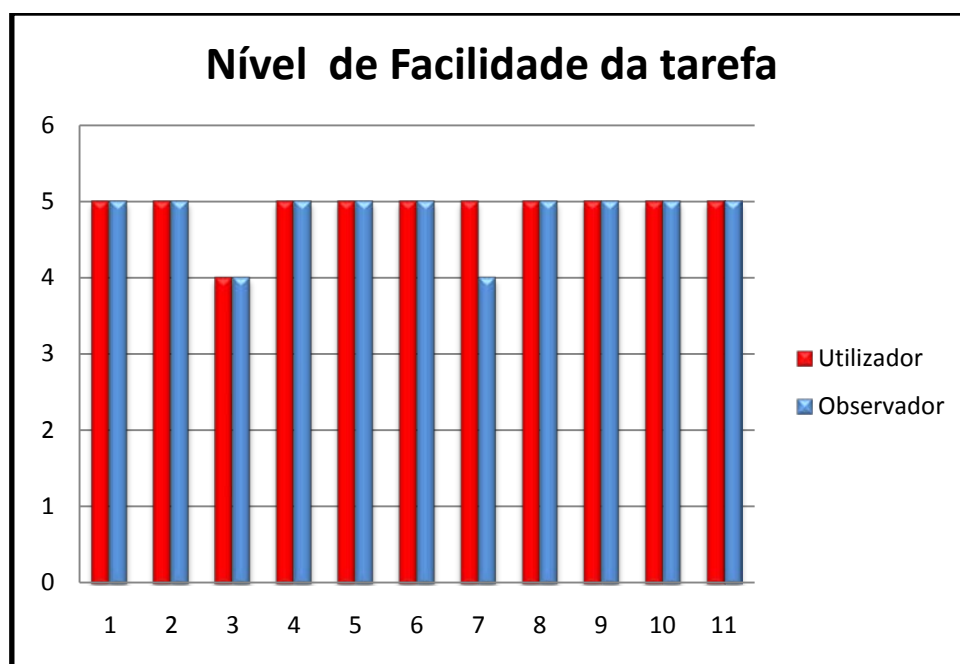


Figura 33 - Nível de facilidade por tarefa

No final do teste de usabilidade foi também pedido aos utilizadores que respondessem a um questionário que teve como objectivos obter informação acerca da posição dos utilizadores relativamente à sua opinião geral sobre a aplicação, opinião sobre aspectos específicos e sugestões de melhoramento.

Encontram-se na Tabela 12 a mediana de valores sobre a opinião dos utilizadores acerca de aspectos gerais da aplicação. A média é feita tendo em conta que o valor **1** corresponde à posição **Discordo totalmente** e **5** a **Concordo totalmente**.

É fácil orientar-me na aplicação	5
Encontro facilmente o que procuro	5
A aplicação é lenta	1
Aplicação é agradável de utilizar	5
A aplicação tem algumas características irritantes	1
Existe consistência na disposição e nos conteúdos apresentados	4
Sinto necessidade de ajuda em algumas funcionalidades	2
A utilização da aplicação requer conhecimentos mais aprofundados ou experiência	2

Tabela 11 - Valores acerca de aspectos gerais da aplicação

Na Tabela 13 encontram-se as medianas dos valores atribuídos às questões sobre a opinião dos utilizadores acerca de aspectos específicos da aplicação.

Os botões têm o tamanho certo	4
O tamanho dos caracteres no ecrã torna-os fáceis de ler	5
A informação mais importante possui um bom destaque	5
A quantidade de informação apresentada por cada ecrã é adequada	5
A disposição da informação no ecrã é adequada	5
Os ícones apresentados são intuitivos	5
O aspecto gráfico é atractivo	5
É fácil navegar na aplicação	5
Sinto-me confortável com o tamanho do ecrã	5

Tabela 12 - Valores acerca de aspectos específicos da aplicação

Analisando as tabelas, concluiu-se que a opinião geral relativamente aos aspectos gerais, bem como aos aspectos físicos, é bastante positiva.

Foi dada a possibilidade aos utilizadores de darem no final do questionário, sugestões para melhoramento da aplicação em geral. As sugestões dadas têm que ver com algumas das tarefas em que os utilizadores cometeram alguns erros na sua execução (tarefa 3 e 7), sendo apresentadas a seguir algumas:

- **Possibilidade de outro tipo de ordenação na listagem das dissertações**

Neste momento apenas está implementada a ordenação por nome do orientador. A possibilidade de ordenar a lista de dissertações por título ou curso facilita a tarefa de procura.

- **Botão com a funcionalidade de ver o perfil, com o nome “Ver Perfil”**

O botão com a funcionalidade de ver o perfil do utilizador, depois de este introduzir as suas credenciais, muda o seu texto de “*login*” para o nome do utilizador (como se mostra na Figura 34). No caso dos testes, a alteração era feita de “*login*” para “utilizador1”, isto porque as credenciais introduzidas foram: **E-mail: utilizador1@ua.pt**. A solução sugerida passaria pela alteração do texto, de “*login*” para “Ver Perfil” ou “Perfil”. Este problema surgiu talvez porque o utilizador introduziu credenciais de teste e não o seu utilizador universal, não tendo imediatamente a percepção de que o botão “utilizador1” tinha a funcionalidade de mostrar o perfil.



Figura 34 - Antes e depois de efectuar *login*

- **Possibilidade de voltar ao menu principal através da página de perfil**

Quando o utilizador se encontra em qualquer área do menu principal, pode sair dessa área utilizando a barra de cabeçalho (*breadcrumbs*) ou o botão “Voltar” para ir

para a página anterior. No caso do botão que tem como objectivo mostrar a página de perfil, encontra-se no rodapé da aplicação, e portanto não faz parte do menu principal. Por este motivo, quando se acede à página de perfil, é necessário seleccionar o botão “Fechar perfil” para sair, não havendo outra alternativa (como mostrado na Figura 35). No entanto, alguns utilizadores sugeriram que seria mais fácil permitir sair da aplicação também através do cabeçalho, o que seria mais consistente tendo em conta que é o meio de navegar entre as áreas presentes no menu principal.



Figura 35 - Diferença entre como sair de uma área do menu principal e sair da área de perfil

- **Efectuar *logout* sem ter que ir à página de perfil.**

A única forma de fazer *logout* da aplicação, é entrando na página de perfil. Aí o botão, antes, com o nome “utilizador1”, altera para “*logout*”. Alguns utilizadores referiram que apesar de terem percebido antes como se fazia *logout*, acharam que seria melhor, em vez de apenas um botão com o nome do utilizador, ter dois botões, um com o nome do utilizador e outro, sempre presente, para fazer *logout*.

4.1.3 Conclusões

Apesar dos pequenos erros cometidos e das sugestões dadas, no geral, os resultados são bastante satisfatórios. Não existem problemas apontados como graves, pelos avaliadores que realizaram a avaliação heurística, nem pelos utilizadores que participaram nos testes de usabilidade, para serem resolvidos urgentemente. Durante a

realização dos testes, a observadora foi anotando comentários dos utilizadores acerca da aplicação. Comentários esses, bastante positivos e de aprovação da aplicação. Na sua maioria, os utilizadores referiram que utilizariam a aplicação no seu dia-a-dia. Em suma, os utilizadores sentiram-se bastante à vontade e satisfeitos com a aplicação.

No entanto, é de notar que os testes foram realizados com apenas sete utilizadores, sendo este aspecto uma limitação deste estudo. Para se puderem obter conclusões mais definitivas seria necessário realizar um estudo mais abrangente, eventualmente envolvendo a utilização da aplicação em condições mais reais. Para que tal se consiga fazer é necessário, por exemplo, a divulgação da aplicação de teste ao Departamento, para que qualquer aluno utilize a aplicação voluntariamente num contexto real e com a possibilidade de utilização por um tempo significativo.

5 Conclusões gerais

A principal motivação deste trabalho foi de obter conhecimento na área do desenvolvimento móvel, assim como a familiarização com o meio envolvente. Para que se conseguisse aprofundar os conhecimentos nesta área foi necessário realizar um estudo sobre o tema desta dissertação, analisando os constrangimentos inerentes à implementação de soluções para este tipo de dispositivos.

Este trabalho foi iniciado com o estudo sobre o contexto em que o projecto se inseria, nomeadamente o estado actual da arte na área dos dispositivos móveis. Possibilitando assim, compreender as soluções já existentes e o que surgiu de novo que enriqueceu a matéria na área.

Foi feita uma análise da plataforma *DSD Web*, mais concretamente à parte *Front Office* do utilizador aluno, identificando os objectivos, funcionalidades existentes bem como as características relevantes para o desenvolvimento deste trabalho, nomeadamente no que diz respeito ao design.

No que diz respeito à implementação do protótipo funcional da aplicação *DSD-mobile*, realizou-se primeiramente, a análise de requisitos do utilizador de forma a identificar as suas exigências e necessidades perante um sistema deste tipo. Esta análise foi realizada com sucesso, dentro dos parâmetros considerados suficientes pois, obteve-se aproximadamente 20% de respostas o que permitiu tirar algumas conclusões acerca das necessidades dos utilizadores e desvendar as barreiras à introdução de novas tecnologias numa instituição de ensino. Tendo em conta os requisitos identificados, a análise da plataforma existente e o estudo efectuado acerca do desenvolvimento *Web* para dispositivos móveis, partiu-se então para a implementação do protótipo funcional.

Foi implementada a interface da aplicação, não tendo sido totalmente operacionalizada devido aos contratempos que foram surgindo ao longo do trabalho, nomeadamente devido à falta de conhecimentos aprofundados nas tecnologias envolvidas. Assim, considera-se que ficará em aberto a plena integração com o sistema

DSD existente, o que teria permitido testar mais concretamente a utilidade da aplicação na actividade diária dos utilizadores.

No entanto, o protótipo efectuado foi submetido a um teste de usabilidade com utilizadores da plataforma *DSD Web*. Da análise dos resultados obtidos na avaliação concluiu-se que a aplicação implementada agradou bastante aos utilizadores. Esta afirmação é suportada, não só, pelos comentários positivos durante o teste bem como pela análise dos guiões e questionários preenchidos durante a sessão.

Assim, apesar do carácter embrionário da aplicação conclui-se que, o *DSD-mobile* representa uma mais-valia para o Departamento de Electrónica, Telecomunicações e Informática, uma vez que permite dinamizar e aproximar o acompanhamento das informações, por parte dos alunos, atenuando assim a habitual exigência de um computador.

Este trabalho permitiu dar a conhecer as potencialidades desta tecnologia, antes de ser efectivamente implementada podendo prevenir problemas capazes de colocar em risco a viabilidade da versão final.

5.1 Trabalho Futuro

A constante evolução na área do desenvolvimento para dispositivos móveis tem feito com que os utilizadores exijam cada vez mais potencialidades em relação aos diversos tipos de equipamento disponíveis. Devido a tais exigências, é fundamental que uma aplicação *Web* seja desenvolvida da melhor forma possível respondendo a esses mesmos requisitos.

Ao longo deste trabalho, encontraram-se algumas limitações, assim como oportunidades de trabalho futuro. Foram-se ouvindo opiniões e sugestões de implementação, que na opinião da autora foram bastante construtivas e tiveram um contributo extremamente importante para o desenvolvimento deste trabalho. No entanto, o resultado final deste trabalho pode ser melhorado e nesse sentido, apresentam-se, nesta secção, algumas sugestões para trabalho futuro com vista a responder às mais variadas necessidades, não só dos utilizadores, mas também do sistema.

Assim sendo, considera-se necessário o desenvolvimento de todas as funcionalidades de forma a possibilitar-se a sincronização da informação entre o dispositivo móvel e a base de dados central do DSD *Web*, mantendo sempre a informação disponível e actualizada. De facto, com mais funcionalidades disponíveis, a aplicação móvel desenvolvida poderia evoluir ao ponto ficar preparada para receber constantemente todas as actualizações, e enviar as modificações em tempo real. Existem bastantes funcionalidades, ainda por implementar, que podem complementar a funcionalidade geral do DSD quando acedido através de um dispositivo móvel. A implementação da interface dos docentes é um exemplo que para alguns professores pode vir a ser indispensável. Apesar de no inquérito inicial não se mostrarem receptivos, o facto de depois de experimentarem uma vez pode fazer toda a diferença.

Optou-se inicialmente por uma aplicação *Web*, no entanto, uma aplicação nativa tem também as suas vantagens nomeadamente a integração da aplicação com o sistema do dispositivo utilizado. Por exemplo, o serviço de localização para integração na funcionalidade da área “Salas” existente na aplicação seria bastante útil, mas para isso, é necessário fazer uma análise mais cuidada relativamente às implicações dos serviços dependentes do sistema do dispositivo.

A agregação de portais, como o da biblioteca da Universidade, para pesquisa de arquivos bibliográficos e o portal Paco para controlo de faltas, pautas e inscrições nos horários. Esta, foi uma sugestão deixada por alguns alunos inquiridos inicialmente, no entanto, a sua integração não foi possível devido ao tempo limitado para a realização deste projecto e às permissões que seriam indispensáveis por parte de quem gere estes serviços. Também a integração de serviços de comunicação em tempo real entre a comunidade académica ou o acesso a redes sociais pode contribuir para difundir a informação, assim como envolver toda a comunidade num serviço apenas, o DSD-mobile.

Outra proposta dada pela autora é a realização de testes da aplicação com mais utilizadores, no seu contexto de vida real, dando-lhes mais tempo para experimentarem e avaliarem a aplicação. Só assim será possível tirar-se conclusões mais definitivas pois o utilizador faz uso da aplicação no seu dia-a-dia, acedendo à informação quando dela necessita.

Para finalizar, propõe-se a continuação deste projecto estruturando o código existente numa lógica SOA (*Service Oriented Architecture*), ou seja, através de *webservices*. Esta camada de abstracção seria vantajosa em alguns aspectos, nomeadamente na diminuição significativa, tanto de código como de tempo de implementação; na obtenção de dados independentemente da linguagem de implementação utilizada; tornaria toda a implementação mais versátil na medida em que a partir do momento em que a informação é disponibilizada em *webservices* podem ser criadas aplicações distintas que “bebem” dados de um repositório só. Assim, o processamento da informação passaria a ser centralizado pois as aplicações acabam por ser apenas motores de tratamento dos dados retomados pelos *webservices*.

6 Referências bibliográficas

- Abran, A., J. Moore, et al. (2004). Guide to the Software Engineering Body of Knowledge. California, IEEE Computer Society.
- Apers, C. (2010). "Virtual Panel: The State of the Art in Mobile Web Application Development." Consultado em Outubro 2010, Disponível em <http://www.infoq.com/articles/mobile-web-development>.
- Appelquist, D. (2006). What is "Mobile 2.0" (Beta). Dan's Blog (2.0) - Musings on technology, the Web, mobility and beyond.
- Ballard, B. (2007). Designing the Mobile User Experience. England, Wiley.
- Bergsten, H. (2003). JavaServer pages, O'Reilly & Associates, Inc. Sebastopol, CA, USA.
- Bina, M., D. Karaiskos, et al. (2007). "Motives and Barriers Affecting the Use of Mobile Data Services". International Conference on the Management of Mobile Business (ICMB 2007). Canada.
- Bitstream, I. (2010). "BOLT Frequently Asked Questions." Consultado em Setembro 2010, Disponível em <http://www.boltbrowser.com/faq.html#boltplat>.
- BluePrints, J. (2010). "Model-View-Controller." Consultado em Outubro 2010, Disponível em <http://www.oracle.com/technetwork/java/mvc-detailed-136062.html>.
- Brown, M. (2008). "IIS vs. Apache, Looking Beyond the Rhetoric." Consultado em Outubro 2010, Disponível em http://www.serverwatch.com/tutorials/article.php/10825_3074841_1/IIS-vs-Apache-Looking-Beyond-the-Rhetoric.htm.
- Bruseberg, A. e D. McDonagh-Philp (2001). "New product development by eliciting user experience and aspirations." International Journal of Human Computer Studies 55(4): 435-452.
- Campos, D. E. M. (2008). Distribuição de Serviço Docente: back e front office Web. Departamento de Electrónica, Telecomunicações e Informática. Aveiro, Universidade de Aveiro: 148.
- Cardoso, P. (2009). "Bases de Dados." Consultado em Outubro 2010, Disponível em <http://www.est.ualg.pt/ADEe/disciplinas/ciclo1/ano3/semestre1/basesdados/material/20090925-BDaulas.pdf>.
- Converse, T., J. Park, et al. (2004). PHP5 and MySQL bible, Hungry Minds Inc.
- Corporation, O. (2010). "Why MySQL?" Consultado em Outubro 2010, Disponível em <http://www.mysql.com/why-mysql/>.
- Cotton, J. e P. Commarford. (2005). "Designing Web content for mobile browsers." Consultado em Setembro 2010, Disponível em <http://www.ibm.com/developerworks/library/wi-web/index.html>.
- Damas, L. (2007). SQL - Structured Query Language, FCA.
- Darje, C. e W. Barnett (2008). Build Your Own ASP.NET 3.5 Web Site Using C# & VB. Cambridge, SitePoint Pty. Ltd.
- Davis. (1997). "Application Development Methodology." Consultado em Setembro 2010, Disponível em <http://sysdev.ucdavis.edu/WEBADM/document/rad-purpose.htm>.
- Deacon, J. (2005). "Model-view-controller (MVC) architecture." JOHN DEACON Computer Systems Development, Consulting & Training.
- Dix, A., J. Finlay, et al. (2004). Human Computer Interaction. Harlow, Pearson Education.
- Esposito, N. (2010). "How to Design for Mobile?" Web Design Trends of 2010 Consultado em Outubro 2010, Disponível em <http://www.austin-williams.com/blog/post.cfm/web-design-trends-of-2010>.

- Exame Informática (2007). Venda de PDA cai 40%. Exame Informática, Grupo Impresa.
- Ferreira, C. A. (2010). As aplicações já tomaram conta do seu "smartphone"? Jornal de Negócios Online.
- Firtman, M. (2010). Programming the Mobile Web, O'Reilly Media, Inc.
- Fling, B. (2009). Mobile Design and Development. Sebastopol, Steven Weiss
- Frederick, G. R. e R. Lal (2009). Beginning Smartphone Web Development: Building JavaScript, CSS, HTML and Ajax-based Applications for iPhone, Android, Palm Pre, BlackBerry, Windows Mobile, and Nokia S60. New York, Apress.
- Gaillard, S. (2010). "Web Safe Fonts." Consultado em Setembro 2010, Disponível em <http://ezinearticles.com/?Web-Safe-Fonts&id=3825292>.
- Gartner, I. (2010). "Gartner Says Android to Become No. 2 Worldwide Mobile Operating System in 2010 and Challenge Symbian for No. 1 Position by 2014." Consultado em 18 Setembro 2010, Disponível em <http://www.gartner.com/it/page.jsp?id=1434613>.
- Gatti, A. e A. Jaokar (2009). Open Mobile - Understanding the Impact of Open Mobile: Implications for Telecoms/Devices, Web, Social Networks, Media and Personal Privacy. London, Futuretext.
- Goguen, J. A. e C. Linde (1993). Techniques for Requirements Elicitation. Proceedings of the Requirements Engineering, IEEE Computer Society.
- Gomes, I. (2006). "Soluções web centradas no utilizador." Consultado em Setembro 2010, Disponível em <http://www.ivogomes.com/blog/solucoes-web-centradas-no-utilizador/>.
- Google. (2010a). "Google Chrome to Phone Extension." Consultado em Setembro 2010, Disponível em <http://code.google.com/p/chrometophone/>.
- Google. (2010b). "Google Projects for Android." Consultado em Setembro 2010, Disponível em <http://code.google.com/intl/pt-PT/android/>.
- Hiner, J. (2000). "The Web server software choice: iPlanet vs. Apache vs. IIS." Consultado em Outubro 2010, Disponível em http://articles.techrepublic.com.com/5100-10878_11-1034514.html.
- Holtzblatt, K. (2005). Designing for the mobile device: experiences, challenges, and methods, Communications of the ACM.
- Hosbond, J. H. (2005). Mobile systems development: challenges, implications and issues, Springer Publishing.
- Informa Telecoms & Media. (2007). "Mobile Advertising & Marketing USA '08." Consultado em Junho 2010, Disponível em http://shop.informatm.com/marlin/30000001001/MARKT_EFFORT/marketingid/20001615500.
- ISO (1998b). ISO 9241 /11 - Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability., International Organization for Standardization.
- ISO (1999). ISO 13407: Human-centred design processes for interactive systems, International Standards Organisation.
- Jaokar, A. (2006) "Ajit Jaokar's Mobile Web 2.0 Blog: What is "Mobile Web 2.0"?" Web 2.0: Article.
- Johnson, D. (2007). "Color Psychology - Do different colors affect your mood?" Infoplease.
- JQuery, P. (2010). "jQuery - write less, do more." Consultado em Outubro 2010, Disponível em <http://jquery.com/>.
- Kayne, R. (2010). "What is a Web Browser?" Consultado em Setembro 2010, Disponível em <http://www.wisegeek.com/what-is-a-web-browser.htm>.
- Kennedy, B. e C. Musciano (2002). HTML & XHTML: The Definitive Guide. Sebastopol, O'Reilly & Associates, Inc.
- Lee, V., H. Schneider, et al. (2005). Aplicações móveis: arquitetura, projeto e desenvolvimento. São Paulo, Pearson.

- Leffingwell, D. e D. Widrig. (2003). "Managing Software Requirements:A Use Case Approach." Consultado em, Disponível.
- LIEN, J. D. (2010). "Opera Mini 5.1 for Android Released." Consultado em 5 de Agosto 2010, Disponível em <http://my.opera.com/portalnews/blog/2010/07/14/opera-mini-5-1-for-android-released>.
- Lindberg, H. e P. Rydin (2002). Model View Controller. United States, New York. **us 2002/0143800 a1**.
- MacDonald, M. (2006). Beginning ASP.NET 2.0 in C# 2005: From Novice to Professional, Apress.
- Maguire, M. (2001). Methods to support human-centred design, *International Journal of Human-Computer Studies*.
- Maguire, M. e N. Bevan (2002). User requirements analysis. Canada.
- Mandel, T. (1997). Elements of user interface design. New York, John Wiley & Sons.
- Marktest, B. d. T. (2010). "Posse de telemóvel acima de 90%." Consultado em 1 de Julho de 2010, Disponível em <http://www.marktest.com/wap/a/n/id~14db.aspx>.
- Marques, N. (2010). "O abc da revolução tecnológica." Life & Style Consultado em Setembro 2010, Disponível em <http://aeiou.expresso.pt/o-abc-da-revolucao-tecnologica=f556031>.
- Mayhew, D. (1999). The usability engineering lifecycle. San Francisco (CA), Morgan Kaufman.
- McHugh, S. (2010). "Image Posterization." Consultado em Setembro 2010, Disponível em <http://www.cambridgeincolour.com/tutorials/posterization.htm>.
- Mehta, N. (2008). Mobile Web Development. Birmingham, B27 6PA, UK, Packt Publishing Ltd.
- Microsoft, C. (2009a). "Expect the unexpected." Consultado em Outubro 2010, Disponível em <http://www.microsoft.com/net/dotnetstories/default.aspx>.
- Microsoft, C. (2009b). "Microsoft Ajax Content Delivery Network." Consultado em Outubro 2010, Disponível em <http://www.asp.net/ajaxlibrary/cdn.ashx>.
- Microsoft, C. (2009c). ".NET Framework Overview." Consultado em Outubro 2010, Disponível em <http://www.microsoft.com/net/overview.aspx>.
- Microsoft, C. (2010a). "ASP.NET MVC Overview (C#)." Consultado em Outubro 2010, Disponível em <http://www.asp.net/mvc/tutorials/asp-net-mvc-overview-cs>.
- Microsoft, C. (2010b). "Microsoft Silverlight." Consultado em Outubro 2010, Disponível em <http://www.silverlight.net/>.
- Microsoft, C. (2010c). "Microsoft typography - ClearType Tuner." Consultado em Setembro 2010, Disponível em <http://www.microsoft.com/typography/cleartype/tuner/tune.aspx>.
- Microsoft, C. (2010d). ".NET Framework Developer Center." Consultado em Outubro 2010, Disponível em <http://msdn.microsoft.com/pt-br/netframework/default.aspx>.
- Microsoft, C. (2010e). "Product Information." Consultado em Outubro 2010, Disponível em <http://www.microsoft.com/sqlserver/>.
- Microsoft, C. (2010f). "What is IIS?" Consultado em Outubro 2010, Disponível em <http://www.iis.net/overview>.
- Mitchell, P. P. (2007). A step-by-step - Guide to Usability Testing. Lincoln, Nebraska, iUniverse.
- Mota, L. S. (2009). DSD: interfaces e interações. Departamento de Electrónica, Telecomunicações e Informática. Aveiro, Universidade de Aveiro.
- MOZILLA, D. N. (2010a). "About JavaScript." Consultado em Outubro 2010, Disponível em https://developer.mozilla.org/en/About_JavaScript.
- Mozilla, F. (2010b). "Sync your desktop and mobile browsing." Consultado em Setembro 2010, Disponível em <http://www.mozilla.com/en-US/mobile/features/#sync>.
- Niederauer, J. (2007). Web Interativa com Ajax e PHP, Novatec.
- Nielsen, J. (1993). Usability engineering. San Francisco, USA, Morgan Kaufmann – Academic Press.

- Nielsen, J. (2000a). Designing web usability: The practice of simplicity, New Riders Publishing.
- Nielsen, J. (2000b). "Ten Usability Heuristics." Consultado em 28 de Janeiro, Disponível em http://www.useit.com/papers/heuristic/heuristic_list.html.
- Nielsen, J. (2005). "Ten Usability Heuristics." Consultado em Setembro 2010, Disponível em http://www.useit.com/papers/heuristic/heuristic_list.html.
- Nokia. (2010). "Nokia N900 - Device Details." Consultado em Setembro 2010, Disponível em http://www.forum.nokia.com/Devices/Device_specifications/N900/.
- Nuseibeh, B. e S. Easterbrook (2000). Requirements engineering: a roadmap. Proceedings of the IEEE International Conference on Software Engineering. Limerick, Ireland, ICM Press.
- O'Reilly, T. (2005a). "Web 2.0: Compact Definition?" Consultado em Setembro 2010, Disponível em <http://radar.oreilly.com/archives/2005/10/web-20-compact-definition.html>.
- O'Reilly, T. (2005b). "What Is Web 2.0." Consultado em Setembro 2010, Disponível em <http://oreilly.com/web2/archive/what-is-web-20.html>.
- Oliveira, E., M. Pereira, et al. (2005). "Compreensão de aplicações web: o processo e as ferramentas."
- Opera Software. (2010). "Opera Mini & Opera Mobile." Consultado em 5 Agosto 2010, Disponível em <http://www.opera.com/mobile/specs/>.
- Oracle, C. (2010a). "Manual de Referência do MySQL 4.1." Consultado em Outubro 2010, Disponível em <http://dev.mysql.com/doc/refman/4.1/pt/index.html>.
- Oracle, C. (2010b). "Oracle Database 10g Express Edition Datasheet." Consultado em Outubro 2010, Disponível em <http://www.oracle.com/technetwork/database/express-edition/overview/index.html>.
- Oracle, C. (2010c). "Oracle Java Development Tools Support." Consultado em Outubro 2010, Disponível em https://shop.oracle.com/pls/ostore/f?p=ostore:product:7510199057952678::NO:RP,3:P3_LPI:14756161144420584792660#.
- Orenstein, D. (2000). QuickStudy: Application Programming Interface (API). Computerworld. Framingham, Computerworld Research.
- Pereira, B. F. (2010). "Aplicações mobile ou sites mobile - o que é melhor?" Consultado em Setembro 2010, Disponível em http://imasters.uol.com.br/artigo/17070/desenvolvimento/aplicacoes_mobile_ou_sites_mobile_o_que_e_melhor/.
- Preece, J., Y. Rogers, et al. (1994). Human-Computer Interaction. England, Addison-Wesley.
- RIM. (2010a). "BlackBerry Smartphone Browser Guide - Instant Information." Consultado em Setembro 2010, Disponível em <http://demos.blackberry.com/browser/na/us/gen/>.
- RIM, L. (2010b). "Research In Motion Company." Consultado em Setembro 2010, Disponível em <http://www.rim.com/company/>.
- Romaní, C. C. e H. P. Kuklinski (2007). Planeta Web 2.0 - Inteligencia colectiva o medios fast food.. Universitat de Vic. Flacso México. Barcelona / México DF, Grup de Recerca d'Interaccions Digitals.
- Santos, E. V. d. e S. C. S. d. Santos. (2010). "SGBD – Recursos para armazenamento e gerenciamento de documentos." Consultado em Outubro 2010, Disponível em <http://www.devmedia.com.br/post-15675-sgbd-recursos-para-armazenamento-e-gerenciamento-de-documentos.html>.
- Schafer, S. M. (2010). HTML, XHTML, and CSS BIBLE. Indianapolis, Wiley Publishing, Inc.
- Schmidt, E. (2006). "Let more of the world access the internet." Consultado em Setembro 2010, Disponível em <http://www.ft.com/cms/s/2/fbc969aa-e8f2-11da-b110-0000779e2340.html>.
- Sekaran, U. (2003). Research Methods for Business: A Skill-Building Approach USA, John Wiley & Sons Inc.

- Sencha, I. (2010). "Sencha Touch - Overview." Consultado em Outubro 2010, Disponível em <http://www.sencha.com/products/touch/>.
- Sharma, C. (2008). "Mobile Industry Predictions 2008." Consultado em Agosto 2010, Disponível em <http://www.chetansharma.com/MobilePredictions2008.htm>.
- Shneiderman, B. (1998). Designing the user interface: strategies for effective human-computer interaction. California, Addison Wesley Longman.
- Shneiderman, B. e C. Plaisant (2005). Designing the user interface: strategies for effective humancomputer interaction. Boston (MA), Pearson.
- Shrine, N. (2004). "Developing JSPs and Servlets with Netbeans." Consultado em Outubro 2010, Disponível em <http://supportweb.cs.bham.ac.uk/documentation/java/servlets/netbeans-webapps/netbeans-webapps.pdf>.
- Silberschatz, A., P. B. Galvin, et al. (2010). Operating System Concepts Essentials. USA, John Wiley & Sons.
- Silva, M. S. (2010). JQuery - A Biblioteca do Programador JavaScript. São Paulo, NOVATEC
- Skov, J. K. e. M. B. (2003) "Evaluating the usability of a mobile collaborative system: exploring two different laboratory approaches."
- Skyfire. (2010a). "Now there's a Smart Browser for Smart phones." Consultado em Setembro 2010, Disponível em <http://www.skyfire.com/product>.
- Skyfire. (2010b). "Skyfire FAQ for iPhone." Consultado em Setembro 2010, Disponível em <http://www.skyfire.com/support/faq/skyfire-faq-for-iphone>.
- Snell, S. (2009). "Mobile Web Design Trends For 2009." Consultado em Outubro 2010, Disponível em <http://www.smashingmagazine.com/2009/01/13/mobile-web-design-trends-2009/>.
- Sommerville, I. (2007). Software Engineering. Harlow, Addison-Wesley.
- Sousa-Santos, B., L. Teixeira, et al. (2007). Avaliação de usabilidade de alguns aspectos de um serviço integrado no portal ua.pt: My.UA., Revista do DETUA.
- Standards, P. W. (2010). "Working together for standards The Web Standards Project - HTML Versus XHTML." Consultado em Outubro 2010, Disponível em <http://www.webstandards.org/learn/articles/askw3c/oct2003/>.
- Stankovic, S. e D. Simic (2010). "Applying MVC and PAC patterns in mobile applications." Journal of Computing 2(1).
- Steinbock, D. (2005). The Mobile Revolution. London, United Kingdom, Kogan Page
- Stolfi, A. (2002, Setembro 2010). "Serifas." Variáveis tipográficas Consultado em Setembro 2010, Disponível em <http://www.finetanks.com/referencia/serifas.php>.
- Sun. (2002). "Designing Enterprise Applications with the J2EETM Platform." 4. Consultado em Outubro 2010, Disponível em http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e/web-tier/web-tier5.html.
- Surowiecki, J. (2005). "The Wisdom of Crowds." Consultado em Setembro 2010, Disponível em <http://www.randomhouse.com/features/wisdomofcrowds/>.
- Teixeira, L. d. C. (2008). Contribuições para o Desenvolvimento de Sistemas de Informação na Saúde: Aplicação na Área da Hemofilia. Departamento de Economia, Gestão e Engenharia Industrial. Aveiro, Universidade de Aveiro: 320.
- The Apache Software Foundation. (2010). "Apache - HTTP Server Project." Consultado em Outubro 2010, Disponível em <http://httpd.apache.org/>.
- The PHP Group. (2010). "O que é PHP?" Consultado em Novembro 2010, Disponível em http://www.php.net/manual/pt_BR/intro-what-is.php.
- Tittel, E. e J. Noble (2008). HTML, XHTML & CSS For Dummies. Indianapolis, Wiley Publishing, Inc.

- Trulock, V. (2008). "Understanding hci - Storyboards." Consultado em Outubro 2010, Disponível em http://ilikecake.ie/hci/env_storyboards.htm.
- Tryfos, P. (1996). Sampling Methods for Applied Research: Text and Cases. New York, John Wiley & Sons.
- Ullman, C. e L. Dykes (2007). Beginning Ajax. Indianapolis, Wiley Publishing, Inc.
- Usability.gov. (2010a). "Card Sorting." Consultado em Setembro 2010, Disponível em http://www.usability.gov/methods/design_site/cardsort.html.
- Usability.gov. (2010b). "User-Centered Design." Consultado em Setembro 2010, Disponível em <http://www.usability.gov/basics/ucd/index.html>.
- Volodarsky, M., O. Londer, et al. (2008). Internet Information Services (IIS) 7.0, Microsoft Press.
- W3C. (2005). "Document Object Model (DOM)." Consultado em Outubro 2010, Disponível em <http://www.w3.org/DOM/>.
- W3C. (2010a). "Tim Berners-Lee." Consultado em Outubro 2010, Disponível em <http://www.w3.org/People/Berners-Lee/>.
- W3C. (2010b). "XHTML2 Working Group Home Page." Consultado em Outubro 2010, Disponível em <http://www.w3.org/MarkUp/>.
- W3C, R. (2002). "XHTML™ 1.0 The Extensible HyperText Markup Language." Consultado em Outubro 2010, Disponível em <http://www.w3.org/TR/xhtml1/>.
- W3Schools. (2010a). "ASP Tutorial." Consultado em Outubro 2010, Disponível em <http://www.w3schools.com/asp/default.asp>.
- W3Schools. (2010b). "JavaScript Tutorial." Consultado em Outubro 2010, Disponível em http://www.w3schools.com/js/js_intro.asp.
- W3Schools. (2010c). "XHTML Tutorial." Consultado em Outubro 2010, Disponível em <http://www.w3schools.com/xhtml/>.
- Wang, D. (2010). "How does JSP work on IIS?" Consultado em 30 de Julho de 2010, Disponível em <http://blogs.msdn.com/b/david.wang/archive/2005/10/11/how-does-jsp-work-on-iis.aspx>.
- Web Standards Group. (2010). "About Web Standards." Consultado em Setembro 2010, Disponível em <http://webstandardsgroup.org/standards/>.

7 Anexos

7.1 Anexo A

7.1.1 Questionário preliminar para análise de requisitos

Questionário Preliminar - DSD-Mobile

O objectivo deste questionário, no âmbito do projecto de **dissertação DSD-Mobile**, do Mestrado Integrado em Engenharia Electrónica e Telecomunicações, é recolher informações acerca do **perfil dos utilizadores** da plataforma DSD - Distribuição de Serviço Docente da Universidade de Aveiro.

Todos os dados recolhidos serão utilizados apenas para os fins em causa e serão processados com total **confidencialidade e anonimato**.

A sua ajuda é fundamental.

Muito obrigada!

1. Informações Pessoais e Profissionais

1.1 Idade

1.2 Género

☐ Feminino

☐ Masculino

1.3 Estatuto

☐ Aluno

☐ Docente

☐ Funcionário não Docente

☐ Outro: _____

1.4 No caso de ser Funcionário não Docente indique qual a função que desempenha.

2. Informações Educacionais

2.1 Qual o seu grau de escolaridade?

(No caso de ser docente ou Funcionário não Docente, indique o seu grau de formação)

☐ Licenciatura

☐ Mestrado

☐ Doutoramento

☐ Outro: _____

2.2 Qual o curso que frequenta?

(Esta questão destina-se apenas a alunos)

☐ Engenharia Electrónica e Telecomunicações

☐ Engenharia de Computadores e Telemática

☐ Tecnologias e Sistemas de Informação

☐ Outro: _____

2.3 Qual o ano que frequenta?

(Esta questão destina-se apenas a alunos)

☐ Primeiro

☐ Segundo

☐ Terceiro

- ☐ Quarto
- ☐ Quinto
- ☐ Outro: _____

3. Utilização de dispositivos móveis

3.1 Que tipo de telemóvel possui?

- ☐ Telemóvel simples sem acesso à internet
- ☐ Telemóvel simples com acesso à internet
- ☐ PDA (Personal Digital Assistant)
- ☐ Smartphone
- ☐ Não tenho telemóvel
- ☐ Outro: _____

3.2 Que funcionalidades do seu telemóvel costuma utilizar?

- ☐ Telefonar
- ☐ Enviar mensagens de texto
- ☐ Consultar o e-mail
- ☐ Consultar serviços ou páginas web
- ☐ Não utilizo telemóvel
- ☐ Outro: _____

3.3 Se indicou "Consultar serviços ou páginas web", na pergunta anterior, indique quais os serviços e páginas web a que mais recorre.

- ☐ Vídeos (Youtube, Vimeo)
- ☐ Música (rádios online, etc)
- ☐ Fotos (Flickr, etc)

- ☐ Redes Sociais (Facebook, hi5, twitter)
- ☐ Pesquisa (Google, Yahoo, etc)
- ☐ Serviços de comunicação em tempo real (Messenger, Skype, etc)
- ☐ Outro: _____

4. DSD Web

4.1 Com que frequência utiliza a plataforma DSD?

- ☐ Todos os dias
- ☐ Regularmente
- ☐ Uma vez por mês
- ☐ Uma vez por semestre
- ☐ Uma vez por ano
- ☐ Nunca

4.2 Conhece todas as funcionalidades da plataforma DSD?

- ☐ Sim, todas
- ☐ A maior parte
- ☐ Apenas as que me interessam
- ☐ Não conheço muito bem
- ☐ Não conheço

4.3 Para que fins utiliza a plataforma DSD Web?

- ☐ Visualização de conteúdos
- ☐ Download de documentos
- ☐ Upload de documentos
- ☐ Fazer inscrições

- ☐ Tarefas de gestão
- ☐ Não utilizo
- ☐ Outro: _____

5. DSD-mobile

(Imagine que existe uma aplicação web para dispositivos móveis, na qual são integradas as mais relevantes funcionalidades da plataforma DSD Web. Assim sendo, responda às questões seguintes relacionadas com essa aplicação mobile)

5.1 Acha que seria útil acompanhar as actualizações da plataforma DSD através do seu telemóvel?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Indiferente

5.2 Que funcionalidades acha pertinente utilizar através do seu telemóvel?

- ☐ Ver horários
- ☐ Ver dissertações
- ☐ Ver colocação nas opções
- ☐ Ver apresentações
- ☐ Ver notícias
- ☐ Efectuar escolhas de opções
- ☐ Mostrar interesse nas dissertações
- ☐ Ver interesse nas dissertações
- ☐ Ver quais as salas e os professores em aulas

☐ Efectuar todas as tarefas de gestão

☐ Download de documentos

☐ Upload de documentos

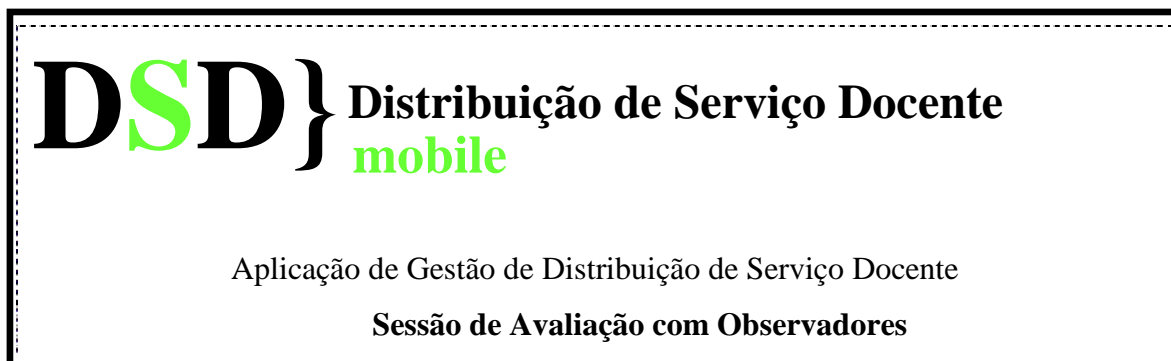
☐ Nenhuma das anteriores

☐ Outro: _____

5.3 Para além das funcionalidades seleccionadas na questão anterior, que funcionalidades acrescentaria ao sistema mobile?

7.2 Anexo B

7.2.1 Lista de tarefas de utilizadores



Este documento apresenta uma lista de tarefas a serem executadas num telemóvel com acesso à Internet durante uma sessão de avaliação na presença de um observador.

Este teste destina-se a avaliar a aplicação e não o utilizador. Por isso não deverá sentir-se pressionado pelo tempo, tarefa incompleta ou outro factor que o possa desconcentrar.

Tente executar cada uma das tarefas de forma descontraída. Sempre que for formulada uma questão por parte do observador, deverá responder oralmente. Pode igualmente formular as questões que achar necessárias ao observador.

Quando tiver efectuado os procedimentos iniciais e se sentir preparado, pode informar o observador e dar início à execução das tarefas. Para isso necessita de abrir no *browser* do telemóvel a aplicação a avaliar.

dsd.av.it.pt/dsdmobile

Depois de terminar cada tarefa, deverá assinalar o grau de facilidade da sua execução usando para o efeito a escala numérica apresentada (onde **1** representa o extremo **Nada Fácil** e **5** o extremo **Muito Fácil**).

Lista de Tarefas

Tarefa 1	<p>Visualize a lista de opções relativas ao 1º Semestre de 2010/2011</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 2	<p>Faça login na aplicação com o seu utilizador universal.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 3	<p>Procure as Dissertações com o título “Dissertação 1” e “Dissertação 2” e mostre o seu interesse nas duas.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 4	<p>Visualize as dissertações nas quais se encontra interessado.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 5	<p>Remova o interesse na “Dissertação 2”.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 6	<p>Inscreva-se nas opções de 5º ano, 1º semestre, com a seguinte ordem de preferências para as três pRIMEiras opções: Interfaces Humano - Computador, Laboratório de Redes, Comunicações Ópticas.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 7	<p>Volte ao Menu. Depois verifique que salas do 2º piso do DETI estão ocupadas neste momento.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 8	<p>Verifique se a docente Maria Beatriz Alves Sousa Santos está disponível neste momento.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 9	<p>Visualize o seu perfil.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 10	<p>Visualize o horário do curso Mestrado Integrado em Engenharia Electrónica e Telecomunicações do 1º semestre, 2º ano.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>
Tarefa 11	<p>Efectue logout para sair da aplicação.</p> <p>-----</p> <p style="text-align: right;">Nada Fácil <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> Muito Fácil</p>

7.2.2 Guião do observador

Tarefa	Solução	Completo a tarefa	Tempo de execução	Cometeu erros	Solicitou ajuda	Grau de facilidade observada (1-Nada Fácil, 5-Muito Fácil)					Obs ·
1	Menu→Listagem	Sim__ Não__	2s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
2	Menu→botão login→introduzir credenciais	Sim__ Não__	5s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
3	Menu→ Dissertações→ Listagem→ Mostrar interesse (Pode utilizar a pesquisa)	Sim__ Não__	4s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
4	Menu→ Dissertações→ Pessoal→ Interessado em	Sim__ Não__	3s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
5	Menu→ Dissertações→ Pessoal→ Interessado em→ Remover interesse	Sim__ Não__	3s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3		5	
6	Menu→ Opções→ Inscrições→ Escolher de 1 a 3	Sim__ Não__	5s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?		2	3	4	5	
7	Menu→ Recursos→ Salas 2ºPiso	Sim__ Não__	2s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
8	Menu→ Recursos→ Docentes (pode utilizar a pesquisa)	Sim__ Não__	3s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
9	Seleccionar o botão Perfil no rodapé da aplicação	Sim__ Não__	2s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
10	Menu→ Horários→ Separador MIEET→ 1º Semestre, 2ºano	Sim__ Não__	1s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	
11	Seleccionar o botão Sair no rodapé da aplicação	Sim__ Não__	1s	Não __ Poucos_ Alguns_	Sim__ Não__ Qual?	1	2	3	4	5	

7.2.3 Questionário final

Questionário Pós – Tarefa

Instruções: Agradecemos a sua colaboração na realização deste estudo, que tem como objectivo avaliar a interface de utilizador da aplicação DSD adaptado a dispositivos móveis e, consequentemente tentar melhorá-la seguindo os critérios de Usabilidade.

A sua colaboração constitui um factor importante para o êxito desta avaliação, por isso solicitamos-lhe o preenchimento deste questionário cujos dados serão utilizados com total anonimato e apenas para fins científicos.

1. Dados Pessoais

Idade: _____

Género: ☐Feminino ☐Masculino

2. Opinião geral sobre o aplicação

Após a utilização da aplicação e tendo em conta a sua avaliação final, assinale com uma cruz o círculo que melhor reflecte a sua opinião em relação à utilização da aplicação. Caso considere que estas quantificações não são aplicáveis, escolha NA.

2.1. Opinião sobre a utilização da aplicação (assinale com uma cruz a opção que melhor corresponde à sua posição).

	Discordo Totalmente				Concordo totalmente	A
É fácil orientar-me na aplicação	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—
Encontro facilmente o que procuro	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—
	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	

A aplicação é lenta		—
Aplicação é agradável de utilizar	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	—
A aplicação tem algumas características irritantes	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	—
Existe consistência na disposição e nos conteúdos apresentados	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	—
Sinto necessidade de ajuda em algumas funcionalidades	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	—
A utilização da aplicação requer conhecimentos mais aprofundados ou experiência anterior	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	—

Se pretender pode deixar aqui outros comentários sobre a utilização da aplicação:

2.2. Opinião sobre aspectos específicos da aplicação (assinale com uma cruz a opção que melhor corresponde à sua posição).

	Discordo Totalmente	Concordo totalmente	NA
Os botões têm o tamanho certo	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>		—
O tamanho dos caracteres no ecrã torna-os fáceis de ler	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>		—
A informação mais importante possui um bom destaque	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>		—

A quantidade de informação apresentada por cada ecrã é adequada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—
A disposição da informação no ecrã é adequada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—
Os ícones apresentados são intuitivos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—
O aspecto gráfico é atractivo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—
É fácil navegar na aplicação	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—
Sinto-me confortável com o tamanho do ecrã	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	—

Se pretender pode deixar aqui outros comentários sobre aspectos específicos da aplicação.

Comentários finais (Se tiver comentários finais pode deixá-los aqui).

Fim do questionário

Muito obrigada pela sua colaboração.